

AD-A190 619

A MULTIPLE MODEL ADAPTIVE TRACKING ALGORITHM AGAINST
AIRBORNE TARGETS(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING

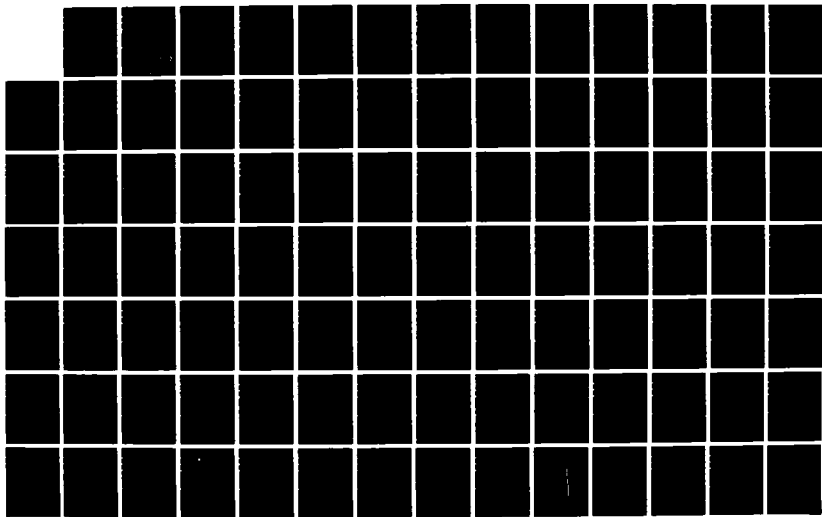
1/4

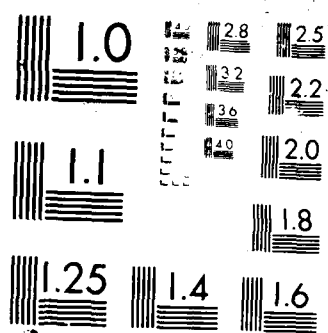
UNCLASSIFIED

T A LEENEY DEC 87 AFIT/OE/ENG/87D-37

F/B 17/5.1

NL





AD-A190 619

DTIC FILE COPY



A MULTIPLE MODEL ADAPTIVE TRACKING
ALGORITHM AGAINST AIRBORNE TARGETS

THESIS

Thomas A. Leeney
Captain, USAF

AFIT/GE/ENG/87D-37

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

DTIC
ELECTE
MAR 28 1988
S E D

Wright-Patterson Air Force Base, Ohio

This document has been approved
for public release and sale in
distribution is unlimited

88 3 24 093

①

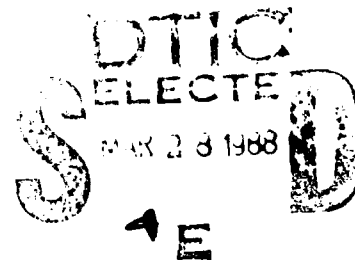
AFIT/GE/ENG/87D-37

A MULTIPLE MODEL ADAPTIVE TRACKING
ALGORITHM AGAINST AIRBORNE TARGETS

THESIS

Thomas A. Leeney
Captain, USAF

AFIT/GE/ENG/87D-37



Approved for public release; distribution unlimited

AFIT/GE/ENG/87D-37

**A MULTIPLE MODEL ADAPTIVE TRACKING
ALGORITHM AGAINST AIRBORNE TARGETS**

THESIS

**Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology**

Air University

**In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering**

Thomas A. Leeney, B.S.E.E.

Captain, USAF

December 1987



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited

Acknowledgments

This effort was part of an ongoing research endeavor to design an effective tracking algorithm for use with the Air Force Weapons Laboratory's high energy laser tracking system. As such, my effort extends the research endeavors of many past Air Force Institute of Technology students. This study investigates the robustness of the tracker to a number of parameter variations.

I would like to express my deepest thanks to my thesis advisor, Dr. Peter Maybeck, for his motivation, guidance and endless insight into the challenges that surfaced during this endeavor. Without his "persistent excitation", this effort would not have reached such a wide scope.

Special thanks are appropriate for my sponsor, the Air Force Weapons Laboratory, especially to Maj. Doug Harnly for providing space structure modeling literature.

To my parents, I express my gratitude for supporting me in all my endeavors throughout the "growing" years.

To Lisa, who did so much to prepare this professional document, I say thank you so much.

Special thanks go to my family, my lovely wife, Angie, and my three beautiful daughters, Brandy, Brenda, and Belinda, for the patience and understanding they portrayed while dad "hacked" away on his computer for countless hours rather than engaging in family activities.

Table of Contents

	Page
Acknowledgments	ii
List of Figures	vi
List of Tables	vii
Notation	ix
Abstract	xiii
I. Introduction	1
1.1 Background	2
1.2 Previous AFIT Research	4
1.3 Objectives	9
1.3.1 Bending and Vibration Phenomena	10
1.3.2 Implementation at 50 Hz	10
1.3.3 Scenario Sensitivity Parameters	10
1.3.4 Target Trajectory Sensitivity	11
1.3.5 Rotating Rectangular FOV	12
1.4 Overview	12
II. Filter Theory	14
2.1 Extended Kalman Filter	14
2.2 Multiple Model Adaptive Filtering ...	17
2.3 Summary	23
III. Truth Model Development	24
3.1 Introduction	24
3.2 Dynamics Model	25
3.2.1 Target States	28
3.2.2 Atmospheric States	30
3.2.3 Mechanical States	33
3.3 Measurement Model	36
3.4 Target Model	39
3.4.1 Target Trajectories	40
3.5 Simulation Space Model	43
3.5.1 Coordinate Frames	43
3.5.2 FLIR Plane Velocity Projection	45
3.5.3 FLIR Plane Target Image Projection	47
3.6 Summary	49
IV. Filter Models	50
4.1 Introduction	50
4.2 Dynamics Models	50
4.2.1 Gauss-Markov Acceleration Model	51

4.2.2	Constant Turn-Rate Dynamics Model	56
4.3	Measurement Model	59
4.3.1	Generating the Template	60
4.3.2	"Pseudo-Measurements" by Enhanced Correlation	63
4.4	Summary	65
V.	Tracking Algorithm	66
5.1	Introduction	66
5.2	Overview of the Tracking Algorithm	66
5.2.1	Gauss-Markov Model Tracker ...	67
5.2.2	Constant Turn-Rate Model Tracker	68
5.3	Field-of-View Processing	70
5.4	Filter Parameters	70
5.5	Tracking Algorithm Statistics	73
5.6	Performance Plots	75
5.7	Greyscales	81
5.8	Summary	83
VI.	Performance Analysis	84
6.1	Introduction	84
6.2	50 Hz Implementation Analysis	84
6.2.1	Gauss-Markov Model	85
6.2.2	Constant Turn-Rate Model	89
6.3	Single Filter Bending/Vibration Analysis	91
6.3.1	Gauss-Markov Model Tracker ...	92
6.3.2	Constant Turn-Rate Model Tracker	94
6.4	MMAF Bending/Vibration Analysis	95
6.4.1	MMAF Benchmark Simulation	96
6.4.2	Effects of Bending/Vibration	98
6.4.3	Increased Bending/Vibration Phenomena	101
6.4.4	Increased Bending/Vibration with Tuning	102
6.4.5	Impact of Bending/Vibration Phenomena	107
6.5	Scenario Sensitivity Analysis	107
6.5.1	Range/Pixel Size Sensitivity	111
6.5.2	High-g Maneuver Sensitivity ..	113
6.5.3	Medium Correlation Time Target	114
6.5.4	Fast Correlation Time Target	116
6.5.5	Pixel Size/Noise Strength Relationship	117
6.6	Target Trajectory Sensitivity	119
6.6.1	Sensitivity to Trajectory #5	119
6.6.2	Ramping Acceleration at Turn Initiation	122
6.7	Compendium	126

VII. Assessments and Recommendations	127
7.1 Introduction	127
7.2 Assessments	127
7.2.1 Practicality of the 50 Hz Sample Frequency	127
7.2.2 Impact of Hardware Bending/ Vibration	128
7.2.3 Filter Sensitivity to Varying Parameters	129
7.2.4 Target Trajectory Results	131
7.3 Investigations of Rotating Rectangular Field-of-View	131
7.4 Recommendations for Further Study ...	136
 APPENDIX A: Derivation of the Bending and Vibration Effects of a Large Space Structure	 139
APPENDIX B: Simulation Program Flow Diagram	149
APPENDIX C: Single Filter Simulations, Benchmark at 30 Hz and 50 Hz	154
APPENDIX D: Single Filter Simulations, Bending and Vibration Analysis	187
APPENDIX E: Multiple Model Adaptive Filter Simulations, Benchmark and Bending/ Vibration Analysis	204
APPENDIX F: Multiple Model Adaptive Filter Simulations, Scenario Sensitivity Analyses	245
APPENDIX G: Multiple Model Adaptive Filter Simulations, Target Trajectory Sensitivity Analysis	287
Bibliography	304
Vita	306

List of Figures

Figure	Page
2.1 Bayesian Multiple Model Filtering Algorithm	20
3.1 The α - δ plane	29
3.2 Response Spectrum for LOS x-direction	34
3.3 FLIR Image Plane Intensity Function for a Three Hotspot Target	38
3.4 Target Hotspot Distribution	40
3.5 Target Trajectories	42
3.6 Geometry to Project Target Inertial Velocity onto the FLIR Image Plane	46
3.7 Target Image Projection Geometry	47
4.1 Enhanced Correlator/Linear Measurement Model	62
5.1 Square Fields-of-View, Filters #1, #2, and #3 ...	71
5.2 Rectangular Fields-of-View, Filters #4 and #5 ...	71
5.3 Performance Plot; GM/MMAF/T2/10-G/QB1; Actual RMS Error vs Filter-Computed RMS Error ...	77
5.4 Performance Plot; GM/MMAF/T2/10-G/QB1; Mean Position Error $\pm \sigma$, at t_1^-	78
5.5 Performance Plot; GM/MMAF/T2/10-G/QB1; Mean Position Error $\pm \sigma$, at t_1^-	79
5.6 Performance Plot; GM/MMAF/T2/10-G/QB1; Mean Centroid Error $\pm \sigma$, at t_1^+	80
5.7 Noise Corrupted FLIR Measurement Array	82
6.1 PSD Plot of a First-Order Gauss-Markov Process ..	110
6.2 Trajectory #2 and #6 Acceleration Profiles	123
7.1 Spatial Domain to R-Theta Transformation	135
A.1 Dominant Vibration Response Spectrum	141

List of Tables

Table	Page
5.1 MMAP Elemental Filters	66
5.2 Gauss-Markov Filter Parameter Values	67
5.3 CTR Filter Parameter Values	69
6.1 Single Filter Benchmark; GM/SNG-S/T2/10-G	88
6.2 Single Filter Simulation; GM/SNG-S/T2/10-G//50 Hz	88
6.3 Single Filter Benchmark; CTR/SNG-S/T2/10-G	90
6.4 Single Filter Simulation; CTR/SNG-S/T2/10-G//50 Hz	90
6.5 Single Filter with Bending/Vibration; GM/SNG-S/T2/10-G/QB1	93
6.6 Single Filter with Bending/Vibration; CTR/SNG-S/T2/10-G/QB1	94
6.7 Multiple Model Adaptive Filter Benchmark; GM/MMAF/T2/10-G	97
6.8 Dominant Elemental Filters; GM/MMAF/T2/10-G ...	97
6.9 MMAP with Bending/Vibration; GM/MMAF/T2/10-G/QB1	99
6.10 Dominant Elemental Filters; GM/MMAF/T2/10-G/QB1	99
6.11 MMAP with Increased Bending/Vibration; GM/MMAF/T2/10-G/QB2	102
6.12 MMAP with Increased Bending and Tuned; GM/MMAF/T2/10-G/QB2/R+4	105
6.13 Scenario Sensitivity Parameters	109
6.14 MMAP Scenario #1 Statistics; GM/MMAF/T2/2-G/T=8,3/Z ₀ =2,000,000	112
6.15 MMAP Scenario #4 Statistics; GM/MMAF/T2/20-G/T=4,1.5	114
6.16 MMAP Scenario #5 Statistics; GM/MMAF/T2/10-G/T=4,.8	115

6.17	MMAF Scenario #8 Statistics; GM/MMAF/T2/20-G/T=4,.4	117
6.18	MMAF Scenario #9 Statistics; GM/MMAF/T2/10-G/T=4,.4/KPT=.002	118
6.19	Trajectory #5 Statistical Data; GM/MMAF/T5/10-G	121
6.20	Trajectory #5 Dominant Filters	121
6.21	Trajectory #6 Statistical Data; GM/MMAF/T6/10-G	125

Notation

\mathbf{a}	: acceleration vector
\mathbf{A}	: filter-computed residual covariance matrix
AR	: aspect ratio of target
\mathbf{b}	: spatially correlated background noise
\mathbf{B}	: input distribution vector
CTR	: constant turn rate
\mathbf{e}	: unit vector
$E(\cdot)$: expected value
EKF	: extended Kalman filter
$\mathbf{f}(\cdot)$: state function in state differential equation
f_x, f_y	: spatial frequencies
\mathbf{F}	: plant dynamics matrix
FLIR	: forward looking infrared
FOV	: field-of-view
$F(\cdot)$: Fourier transform operator
g	: gravity
\mathbf{G}	: driving noise distribution matrix
GM	: Gauss-Markov
$\mathbf{h}(\cdot)$: measurement function
\mathbf{H}	: measurement matrix
Hz	: Hertz, cycles per second
$\mathbf{I}_{n \times n}$: $n \times n$ identity matrix
I	: pixel intensity
IR	: infrared
\mathbf{K}	: Kalman filter gain

K_B : bending/vibration PSD gain parameter
 k_p : proportionality constant (pixels/micro-radian)
 LKF : linear Kalman filter
 LOS : line-of-sight
 MAP : Maximum a Posteriori
 $MMAF$: Multiple Model Adaptive Filter
 \underline{n} : sensor noise
 N : number of Monte Carlo simulation runs
 P_k : hypothesis conditional probability associated with the k-th elemental filter in the MMAF filter structure
 PSD : power spectral density
 \underline{P} : state error covariance matrix
 \underline{Q} : 'strength' of filter driving noise
 \underline{R} : measurement corruption noise covariance
 RMS : root-mean-square
 r : sensor-to-target range
 r_h : sensor-to-target horizontal range
 t : time
 t_0 : initial time
 t_1^- : instant before incorporating measurement
 t_1^+ : instant after incorporating measurement
 \underline{u} : deterministic input vector
 \underline{v} : zero mean, white, Gaussian measurement corruption noise
 \underline{v}_{los} : line-of-sight velocity vector
 \underline{v}_t : target velocity vector
 v_x, v_y : FLIR plane velocity components

w : zero mean, white, Gaussian dynamics driving noise
 x : state vector
 \hat{x} : estimate of true state vector
 \hat{x}_{mmaf} : MMAF estimate of true state vector
 x, y : FLIR plane position coordinates
 z : measurement vector
 α : azimuth angle
 β : elevation angle
 δ : angle between y and α - β plane
 $\delta(\tau)$: Dirac delta function
 $\partial/\partial x$: partial derivative operator
 Δt : sample period = $t_{i+1} - t_i$
 θ : angle between e_α and y los
 σ : standard deviation
 σ^2 : variance (mean squared value) of Gauss-Markov acceleration process
 ζ : damping coefficient
 τ : correlation time of Gauss-Markov process
 Φ : state transition matrix
 ϕ : state transition matrix element
 ω : angular velocity of constant turn
 ω_n : characteristic frequency

SUBSCRIPTS

a : atmospheric jitter
 b : bending/vibration
 c : centroid of target image

d : discrete-time equivalent system
f : filter
n : natural frequency
t : target dynamics
T : truth model
x : FLIR plane x- component
y : FLIR plane y- component
 α : azimuth direction
 β : elevation direction
 \perp : perpendicular to line-of-sight direction

Abstract

This thesis extends the AFIT research directed towards replacing a standard correlation tracker with a Kalman filter bank/enhanced correlation tracker in a high energy laser weapon system. Airborne targets are tracked by a Bayesian multiple model adaptive filtering (MMAF) algorithm, which utilizes an array of infrared sensing detectors as the measurement information for two-dimensional position data. Two different target dynamics models are exercised: a linear, Gauss-Markov acceleration model, and a nonlinear, constant turn-rate model. Performance analyses are accomplished via Monte Carlo simulation techniques. Extending the adaptive potential of the tracking algorithm is of primary emphasis. The effects of bending and vibration of a large space structure on the FLIR's ability to resolve target position is analyzed. Also, a performance comparison/simulation time tradeoff is conducted with the tracking algorithm operating at both 30 Hz and 50 Hz. Sensitivity studies of adaptive responsiveness to varying target trajectories, various filter-assumed correlation times, range to pixel size relationships, and pixel size to filter driving white noise strength relationships are performed. The robustness of the multiple model algorithm is demonstrated by its ability to adapt to scenarios which it had not been previously tuned.

A MULTIPLE MODEL ADAPTIVE TRACKING ALGORITHM AGAINST AIRBORNE TARGETS

I. Introduction

In view of the Strategic Defense Initiative, increased interest has been placed on the laser as a prime candidate for a potential weapon in a space-based defensive system. Because of a laser's ability to transmit energy from the weapon to the target at the speed of light, the need to compute a lead angle, a necessity with ballistic projectiles, is eliminated. However, several factors affect a laser's effectiveness on a given target: the medium in which the beam travels, the power or intensity of the beam, the capability to acquire the target, and the ability to track the target for a sufficient time to neutralize it.

The precision pointing and accurate tracking capabilities are crucial to the development of a laser weapon system. It would not suffice to "paint" the target with laser energy nor is it feasible to have a space-based laser powerful enough that would neutralize a target instantaneously. Limited energy is available in the laser beam, thus motivating the research into highly accurate pointing and tracking systems for space applications.

1.1 Background

The Air Force Weapons Laboratory (AFWL) at Kirtland AFB, New Mexico, is currently developing and testing a high energy laser weapon for use against airborne targets. The current tracking method employs a Forward Looking Infrared Sensor (FLIR) to detect the target passively and maintain a low probability of detection by the target. The FLIR detector plane comprises a 300 x 500 array of pixels, or picture elements, where each pixel can focus detected energy through an angle of 20 micro-radians in two orthogonal directions. The tracking algorithm extracts a "tracking window" from the larger array and processes the detector outputs to align the tracker's field-of-view (FOV). This algorithm detects target offsets from the center of the FOV.

Currently, AFWL employs a correlation tracker to process pointing and tracking sensor information. This tracker compares the previous sample of FLIR information to the present information. Relative position offsets from one data frame to the next are cross correlated and this information update commands the control system to center the target in the field-of-view. This enables the laser to maintain lock on a given target since the FLIR and laser share the same optics.

Since no a priori knowledge of target characteristics is required, the correlation tracking algorithm can be applied to a wide variety of targets; however, it exhibits several limitations. First, the correlation algorithm does

not take advantage of target dynamic characteristics, which have been thoroughly documented. In addition, it is highly susceptible to noise and is not capable of distinguishing between apparent target motion due to signal corruption, as a result of atmospheric distortion, and true target motion. This "jitter" effect (18) could cause a translation in the FLIR image plane and thus result in the laser beam pointing in a direction other than towards the target. Mirror vibration effects and system component bending effects also contribute to the image plane translation phenomena. Another limitation is the time difference between computing the image correlation and the actual time for the gimbal system to engage for pointing. A tracking algorithm that anticipates future target position based upon past and present information could compensate for these limitations.

Since, under the appropriate conditions, it is an optimal linear estimator, the Kalman filter (9) is ideally suited for overcoming these limitations. By modeling different types of targets, various parameters such as size, shape, and acceleration characteristics of each target type would either be known or could be estimated. If statistical characteristics of measurement errors and atmospheric jitter are included, the filter can predict target position more accurately. This prediction, or optimal estimate, reduces tracking errors due to time delays or pointing system lags by allowing the FLIR to anticipate target motion.

1.2 Previous AFIT Research

Over the past eight years, the Air Force Institute of Technology students and staff have generated numerous papers and theses investigating the feasibility and performance benefits of both enhanced correlation/linear Kalman filter tracking and extended Kalman filtering tracking algorithms, all incorporating online adaptations, with the high energy laser pointing and tracking system. Both Netzer (14) and Tobin (19) had summarized their predecessor's endeavors and a modification of their observations follows.

Initially, the study by Mercier (12) demonstrated that the extended Kalman filter (EKF) algorithm outperformed the standard correlation tracker algorithm in the ability to track long range targets modeled as infrared radiation point sources. The FLIR plane image of the target was assumed to have a bivariate Gaussian distribution and was modeled with equal-intensity, circular contours. The four-state filter was predicated on a benign target dynamics model with a first order, zero-mean Gauss-Markov (GM) position model. The intensity distribution due to atmospheric disturbances was initially approximated by a third order shaping filter, and then this was replaced by a first order shaping filter driven by white, Gaussian noise through reduced order modeling. Filter measurement noise was considered to be uncorrelated in both time and space. This tracking algorithm enhanced tracking performance an order of

magnitude better than observed with the correlation tracker in benign scenarios.

Harnly and Jensen (4) incorporated velocity and acceleration estimates to accommodate for the tracking of more maneuverable targets. They extended the FLIR plane image constant-intensity contours to represent an elliptical configuration oriented at specified angles such that the major axis was aligned with the estimated velocity vector (versus circular contours), in addition to incorporating a spatially correlated Gaussian noise model to represent FLIR sensor and background noises. They also incorporated a maneuver detection algorithm and a means of appropriately responding to detected maneuvers by gain changing and modification of the dynamics models. Finally, they modified the algorithm to estimate the target's true size and shape adaptively.

In previous research efforts, the extended Kalman filters were given target intensity function information, specifically that bivariate Gaussian shapes adequately depicted the targets, perhaps requiring some estimation of parameters associated with that function. Research by Singletery (17) and Rogers (16) implemented algorithms which made no a priori assumption about target shape and tested the algorithms against targets with multiple hot-spots and several dynamic orientations. Rogers developed an alternative tracker which used the target shape function as a template for an enhanced correlator. The correlator

produced measurement offsets in the two orthogonal directions from the center of the FOV and fed these measurements as inputs to a linear Kalman filter. A linear Kalman filter rather than a non-linear extended Kalman filter could be utilized since the measurements, or offset distances, were linear functions of the filter states. When analogous performance is achievable, the linear Kalman filter is preferred over the extended Kalman filter due to the lower level of computational loading.

Kozemchak (7) continued the research by implementing the digital signal processing techniques of Rogers' (16) shape identification algorithm for the case of highly dynamic targets. He substantiated that filters based on both the Gauss-Markov target acceleration and constant turn-rate target dynamics exhibited good tracking performance in the presence of dynamic images; these dynamic images corresponded to realistic projections of multiple hot-spot targets onto the FLIR image plane. Tracking performance was also favorable when experiencing drastic image changes inherent in constant g and constant roll-rate maneuvers.

Further research by Millner (13) included a data processing algorithm to generate an estimated intensity function relationship to be utilized as a template in the correlation algorithm, which is the same as Rogers' (16) alternative filter idea, although with a modified dynamics model to address the highly maneuvering target scenario. This method showed that nearly identical tracking

performance was achieved in both the single and multiple hot-spot scenarios. However, inadequate target tracking performance occurred for targets exhibiting maneuvers in excess of five g's.

In an attempt to thwart this limitation, Flynn (3) investigated the possibility of multiple model adaptive filtering (MMAF) techniques for filter implementation. Suizu (18) pursued this effort and successfully implemented the MMAF in addition to demonstrating that it performed well against a wide dynamic range of targets. The model contained a bank of two filters, one tuned for highly maneuvering targets with a FOV of 24 X 24 pixels and the other tuned for benign targets with a FOV of 8 X 8 pixels. Based upon probabilistic weighting, the filter adaptively changed the assumed target dynamics and corresponding FOV array, which resulted in increasing filter performance to allow the tracking of targets varying from benign straight-line trajectories to pulling 20 g's at 20 kilometers. Both the linear Kalman filter/correlation algorithm and the EKF were tested, and both algorithms demonstrated the capability to track targets with trajectories that maneuvered significantly throughout the simulation scenario.

Loving (8) continued the research by adding a filter to the MMAF bank based on an intermediate level of target dynamics, to aid in tracking highly maneuvering targets. She developed a Maximum a Posteriori (MAP) algorithm and com-

pared it to the Bayesian estimator. The MAP algorithm took advantage of the existing MMAF structure and generated state estimates from the one elemental filter with the highest probability of validity, versus probabilistically weighting all elemental filter estimates. The addition of the third elemental filter to the MMAF bank significantly enhanced the tracking ability of highly maneuvering targets. Both the MAP and Bayesian techniques displayed favorable tracking performance against several target maneuvers.

Follow-on research by Netzer (14) expanded the results produced by Loving (8) and employed the linear filter/correlator algorithm developed by Rogers (16) but modified for harsher target dynamics. He investigated the steady state bias errors as a result of the algorithm tracking a high-g, constant turn rate maneuvering target. It was determined that the predominant cause of the bias was due to mismatching of the FLIR azimuth-channel dynamics model and FOV size when a true target maneuver in the elevation direction forced the MMAF to select a harsh dynamics model and wide field-of-view in order to maintain lock on the target. This motivated the investigation of a MMAF based on separate elemental filters that are tuned specifically for harsh maneuvers in either the x- or y-channels, which would enable the tracker to distinguish between x-direction and y-direction maneuvers. This technique would provide the capability for the tracker to expand the FOV in the critical

direction and maintain lock on a maneuvering target while retaining maximum resolution in the non-critical direction.

Most recently, Tobin (19) implemented a constant-turn-rate dynamics model since the zero-mean, first order Gauss-Markov acceleration processes did not adequately describe target dynamics in some cases. Although he demonstrated that the steady-state standard deviation errors were typically smaller in the CTR model, the GM MMAP outperformed the CTR MMAP with respect to the transient characteristics of target maneuvers. Tobin also included rectangular FOV elemental filters tuned specifically for target maneuvers in both the x- and y-directions and ascertained that the tracker maintained lock on the target at the onset of y-direction maneuvers while retaining maximum resolution in the more benign x-direction.

1.3 Objectives

During the past eight years, substantial development has been accomplished on a tracker capable of handling multiple hot-spot targets in which digital signal processing techniques have been employed on FLIR sensor information to identify the underlying target shape. Most recently, adaptation of the field of view to maintain track on harshly maneuvering targets at close ranges has been investigated through multiple model adaptive filtering techniques. Both the correlator tracker and the multiple model adaptive filter tracker have been assessed with respect to several

tracking scenarios. This research is motivated by the fact that a further assessment of the capabilities of these two designs warrants continued investigation in an attempt to extend their applicability. Therefore, the intent is to continue the research effort in developing a viable tracking algorithm by accomplishing the following objectives.

1.3.1 Bending and Vibration Phenomena. Previous research had considered the tracker to be ground-based and the effects of structural bending and vibration were felt to be negligible. This research shall consider the application of the tracker to a space structure and model the effects of bending and vibration. It shall analyze the resulting tracker performance when the filter does not account for these effects through additional states to be estimated. Implementation of bending/vibration phenomena will only be incorporated explicitly into the truth model, while filter retuning may be performed to enhance filter performance without increasing its state dimension.

1.3.2 Implementation at 50 Hz. The current tracking simulation is implemented digitally at a 30 Hz rate and the potential utilization of a 50 Hz tracker implementation is presently being studied by certain industry sectors for space application (5). Therefore, this effort will pursue the implementation of the current simulation software at a 50 Hz rate and analyze the potential benefits.

1.3.3 Scenario Sensitivity Parameters. This research shall perform a sensitivity analysis to assess the

applicability of this tracking algorithm to a possible range of application scenarios. This will encompass developing the relationships to vary pixel size, target range, target type, and noise characteristics. Previous research by both Netzer (14) and Tobin (19) had addressed the sensitivity of the filters to range; however, the intent of this objective is to develop a relationship with the parameters of pixel size, range, noise characteristics, and target type and evaluate tracker limits as a function of these variables.

1.3.4 Target Trajectory Sensitivity. A comparison of the response of the filters to target trajectories other than those for which the filters are tuned is conducted. The five elemental filters in Tobin's multiple model filter algorithm (19) had experienced favorable results against a straight and level trajectory, a constant 2-g, 10-g or 20-g pull-up maneuver with the maneuver initiating in an inertial x-,y- plane parallel to the FLIR plane, a similar constant-g maneuver but ending in a straight trajectory, and a constant turn-rate maneuver initiating in an inertial x-,z- plane perpendicular to the FLIR plane. This effort will consider the implementation of a constant turn-rate maneuver similar to the constant pull-up maneuver initiating in the x-,y- plane and rotate the trajectory by 45° . This maneuver is expected to show the effects of tracking a target whose initial trajectory is as dynamic in the x-direction as it is in the y-direction, rather than singling out one of the

directions for initial dynamic behavior. Another trajectory modification that will be considered is to represent the target's acceleration at the onset of the turning maneuver as an increasing function of time rather than as a physically unrealistic step input (as had been done in the past). The target acceleration rate will be modeled as a ramp function with the maximum attainable acceleration occurring after five sample periods.

1.3.5 Rotating Rectangular FOV. An investigation into the feasibility of implementing a rotating rectangular field of view into the multiple model adaptive filter structure is performed. This field of view is considered to rotate in such a way as to align the "elongated" side with the current estimate of the target's acceleration. The benefits of maintaining lock on a target that can accelerate in arbitrary directions forms the basis of this evaluation.

1.4 Overview

This chapter has presented a synopsis of the effort performed to date in developing a viable tracking algorithm as well as described the intentions of the areas requiring continued study. Chapter II introduces the filter theory concepts that will be needed to understand the algorithm's theoretical foundation. Chapter III provides the development of the various truth models and Chapter IV discusses the aspects of the two possible filter-assumed dynamics models, a Gauss-Markov acceleration model and a constant

turn-rate dynamics model, in addition to the measurement model employed. Chapter V discusses the tracking algorithms along with discussing the tools used for gathering statistics and describing field-of-view processing characteristics. Chapter VI presents the results of the various analyses conducted, and Chapter VII presents the summary of results and suggests areas for further research.

II. Filter Theory

The intent of this chapter is to highlight the mathematical forms of the Extended Kalman Filter (EKF) and the Multiple Model Adaptive Filter (MMAF). A basic understanding and knowledge of linear Kalman filtering techniques is assumed (9,10).

2.1 Extended Kalman Filter

The extended Kalman filter (EKF), for which a complete derivation appears in (10) and is summarized in (19), is an algorithm that allows the states of a nonlinear stochastic system model to be estimated. Unlike the linear Kalman filter (LKF), the extended Kalman filter requires the Taylor series expansion of nonlinear system dynamics and measurement equations at the desired sample intervals. These Taylor series expansions neglect the effect of second and higher order terms, which results in the EKF not complying with the optimality criteria inherent in the linear Kalman filter. The system state relationship for the EKF takes the form of the following nonlinear stochastic differential equation:

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x}(t), \underline{u}(t), t) + \underline{G}(t) \underline{w}(t) \quad (2-1)$$

where:

$\underline{x}(t)$ = n-dimensional state vector

$\underline{u}(t)$ = r-dimensional control input vector

t = time

$\underline{f}(\cdot)$ = n-dimensional nonlinear system plant dynamics vector function

$\underline{Q}(t)$ = $n \times s$ noise distribution matrix

$\underline{w}(t)$ = s -vector zero-mean, white Gaussian noise process of strength $\underline{Q}(t)$; independent of $\underline{x}(t_0)$

As compared to the linear Kalman filter, $\underline{f}(\underline{x}(t), \underline{u}(t), t)$ replaces $[\underline{F}(t)\underline{x}(t) + \underline{P}(t)\underline{u}(t)]$, but similar to the LKF is the assumption that the initial condition is assumed to be a (Gaussian) random n -vector with mean $\hat{\underline{x}}_0$ and covariance \underline{P}_0 . The discrete-time measurements can be represented in the form of the following nonlinear vector function:

$$\underline{z}(t_1) = \underline{h}(\underline{x}(t_1), t_1) + \underline{v}(t_1) \quad (2-2)$$

where:

$\underline{z}(t_1)$ = m -dimensional measurement vector

$\underline{h}[\cdot]$ = $m \times n$ nonlinear vector function relating the measurements to the states

$\underline{v}(t_1)$ = m -vector process of discrete-time zero-mean, white Gaussian noise with covariance $\underline{R}(t_1)$; independent of both $\underline{x}(t_0)$ and $\underline{w}(t)$

The extended Kalman filter measurement update incorporates the measurements via the following relationships:

$$\underline{K}(t_1) = \underline{P}(t_1^-) \underline{H}^T(t_1) \{ \underline{H}(t_1) \underline{P}(t_1^-) \underline{H}^T(t_1) + \underline{R}(t_1) \}^{-1} \quad (2-3)$$

$$\hat{\underline{x}}(t_1^+) = \hat{\underline{x}}(t_1^-) + \underline{K}(t_1) \{ \underline{z}_1 - \underline{h}[\hat{\underline{x}}(t_1^-), t_1] \} \quad (2-4)$$

$$\underline{P}(t_1^+) = \underline{P}(t_1^-) - \underline{K}(t_1) \underline{H}(t_1) \underline{P}(t_1^-) \quad (2-5)$$

where:

$\underline{K}(t_1)$ = Kalman filter gain at time t_1

$\hat{\underline{x}}(t_1)$ = estimate of \underline{x} at time t_1

$\underline{P}(t_1)$ = $n \times n$ state error covariance matrix

(t_1^-) = instant immediately prior to incorporating measurements at time t_1

(t_1^+) = instant immediately after incorporating measurements at time t_1

The $H(t_1)$ matrix is defined as an $m \times n$ matrix of partial derivatives of h with respect to the first argument, evaluated along the nominal trajectory (10:41):

$$H(t_1) = H(\hat{x}(t_1^-), t_1) = \left. \frac{\partial h(x, t_1)}{\partial x} \right|_{x=\hat{x}(t_1^-)} \quad (2-6)$$

Utilizing the results of the measurement update equations for \hat{x} and P as defined in Equations (2-4) and (2-5) as the initial conditions for the propagation cycle from t_1 to t_{i+1} , the following relationships apply:

$$\dot{\hat{x}}(t/t_1) = \hat{f}(\hat{x}(t/t_1), u(t), t) \quad (2-7)$$

$$\begin{aligned} \dot{P}(t/t_1) &= P(t)P(t/t_1) + P(t/t_1)P^T(t) \\ &+ G(t)Q(t)G^T(t) \end{aligned} \quad (2-8)$$

where :

(t/t_1) = estimate at time t given measurements through time t_1

with the initial conditions:

$$\hat{x}(t_1/t_1) = \hat{x}(t_1^+) \quad (2-9)$$

$$P(t_1/t_1) = P(t_1^+) \quad (2-10)$$

The $P(t)$ matrix is defined as the $n \times n$ matrix of partial derivatives of \hat{f} with respect to its first argument, evaluated along the nominal trajectory (10:41):

$$\hat{E}(t) = E(\hat{x}(t/t_1), t_1) = \left. \frac{\partial f(x, y, t)}{\partial x} \right|_{x=\hat{x}(t/t_1)} \quad (2-11)$$

Note that one difference between the linear Kalman filter and the extended Kalman filter is that the equations for propagating and updating the estimation error covariance matrix are coupled to the state estimate relations. This relationship precludes the precomputation of the covariance and gain matrices until the state estimates and measurement values become known at specified sample times.

2.2 Multiple Model Adaptive Filter

In any given real world application, one can predict only to some degree of accuracy all of the possible scenarios of a deterministic target model, thus allowing the idea of uncertainties in the model. The Kalman filter's ability to achieve a high level of performance against a wide variety of situations would require the matching of the uncertain parameters of the dynamics model to the dynamics of the target. However, some of these parameters are changing in time and the system designer does not always have a priori knowledge of the values of parameters that provide optimal performance. One means of overcoming this limitation is to incorporate Multiple Model Adaptive Filter (MMAF) techniques as developed by (8,14,18,19,10).

The MMAF structure can best be developed (10:129,136) by considering a first-order, linear, stochastic

differential equation for a given time-invariant system model of the form:

$$\dot{\underline{x}}(t) = \underline{F}(\underline{a})\underline{x}(t) + \underline{B}(\underline{a})\underline{u}(t) + \underline{G}(\underline{a})\underline{w}(t) \quad (2-12)$$

and noise corrupted, discrete time measurements as:

$$\underline{z}(t_1) = \underline{H}(\underline{a})\underline{x}(t_1) + \underline{v}(t_1) \quad (2-13)$$

where:

$\underline{x}(t)$ = n-dimensional system state vector

$\underline{u}(t)$ = r-dimensional deterministic control vector

$\underline{w}(t)$ = s-dimensional discrete-time white Gaussian, zero-mean noise vector process of strength $\underline{Q}(\underline{a})$

$\underline{z}(t_1)$ = m-dimensional measurement vector

$\underline{v}(t_1)$ = m-dimensional discrete-time white Gaussian, zero-mean noise vector process of covariance $\underline{R}(\underline{a})$

\underline{a} = uncertain and/or dynamic parameter vector

$\underline{F}(\underline{a})$ = n x n system plant matrix

$\underline{B}(\underline{a})$ = n x r input distribution matrix

$\underline{G}(\underline{a})$ = n x s noise distribution matrix

$\underline{H}(\underline{a})$ = m x n vector relating the measurements to the states

To represent a target with K significantly different sets of dynamics characteristics, it is necessary to discretize \underline{a} into a set of K finite vector values, $\underline{a}_1, \underline{a}_2, \dots, \underline{a}_K$. The MMAP consists of a bank of K independent Kalman filters which are processed in parallel, in which each filter is tuned for a specific target dynamics characteristic

determined by the appropriate \underline{a}_k , for $k = 1, 2, \dots, K$. At discrete sample times, the residuals of each elemental filter are used to calculate the conditional probability that \underline{a} assumes the value of \underline{a}_k associated with that particular elemental filter, conditioned on the observed measurement history. This conditional probability, called the hypothesis conditional probability and denoted $p_k(t_i)$, identifies the elemental filter with the greatest probability of best performance at a given time.

The state estimate of the MMAF, $\hat{\underline{x}}_{mmaf}(t_i^+)$, is the probabilistically weighted average of the elemental bank of filters (19:14):

$$\hat{\underline{x}}_{mmaf}(t_i^+) = \sum_{k=1}^K p_k(t_i) \hat{\underline{x}}_k(t_i^+) \quad (2-14)$$

This form of filter structure, as depicted in Figure 2.1, is referred to as the Bayesian MMAF (10:132), where the state estimate consists of weighting all k filters using the hypothesis conditional probability function:

$$p_k(t_i) = \frac{\int \underline{z}(t_i) | \underline{a}, \underline{z}(t_{i-1}) \quad (\underline{z}_i | \underline{a}_k, \underline{z}_{i-1}) \cdot p_k(t_{i-1})}{\sum_{j=1}^K \int \underline{z}(t_i) | \underline{a}, \underline{z}(t_{i-1}) \quad (\underline{z}_i | \underline{a}_j, \underline{z}_{i-1}) \cdot p_j(t_{i-1})} \quad (2-15)$$

where:

$$\int \underline{z}(t_i) | \underline{a}, \underline{z}(t_{i-1}) \quad (\underline{z}_i | \underline{a}_k, \underline{z}_{i-1}) = \frac{\exp \{ \cdot \}}{(2\pi)^{m/2} |\Delta_k(t_i)|^{1/2}} \quad (2-16)$$

$$\{\cdot\} = (-1/2) \underline{\underline{z}}_k^T(t_1) \underline{\underline{\Delta}}_k^{-1}(t_1) \underline{\underline{z}}_k(t_1) \quad (2-17)$$

$\underline{\underline{\Delta}}_k$ = k-th filter's computed residual covariance

$$= \underline{\underline{H}}_k(t_1) \underline{\underline{P}}_k(t_1^-) \underline{\underline{H}}_k^T(t_1) + \underline{\underline{R}}_k(t_1) \quad (2-18)$$

$\underline{\underline{z}}_k(t_1)$ = k-th filter's residual

$$= [\underline{\underline{z}}(t_1) - \underline{\underline{H}}_k(t_1) \hat{\underline{\underline{x}}}_k(t_1^-)] \quad (2-19)$$

and

$\underline{\underline{a}}_k$ = parameter value assumed in the k-th filter

$\underline{\underline{P}}_k(t_1^-)$ = k-th filter's computed state error covariance
before incorporating the measurement at time t_1 .

$\underline{\underline{z}}(t_{1-1})$ = measurement history up to time t_{1-1}

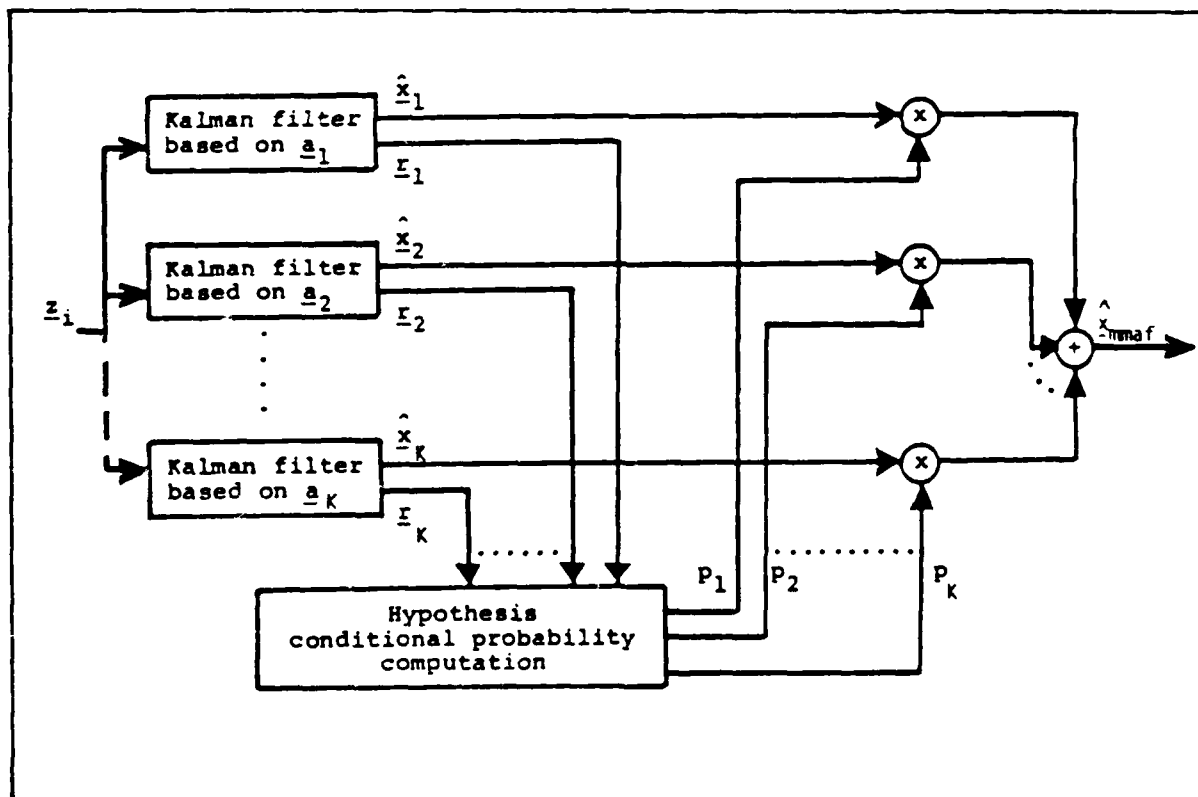


Figure 2.1. Bayesian Multiple Model Filtering Algorithm

As is evident from Equation (2-15), the numerator is the k^{th} filter's product of the previous hypothesis probability and the conditional probability density of the current measurement $z(t_1)$ provided that the k^{th} filter's assumed parameter value a_k and the previous measurement history $z(t_{1-1})$ are known. The denominator is the sum of similar products for all K filters in the bank; this implies that the filter out of the K filters that produces the smallest squared residual relative to the filter-computed residual covariance best matches the real world situation (10:17).

The filter structure of Figure 2.1 conveys that the K filters in the bank process their estimates and residuals in parallel. It is important that the residuals from the "best-matched" filter be distinguishable from those of the mismatched filters. Failure to obtain such distinction can cause inappropriate assignment of large probabilities to filters based on incorrect models, leading to poor performance. To preclude such possible performance degradation, each filter in the bank should be tuned for a unique target trajectory to match its internal dynamics model, since the filter which represents the closest "match" to the true target dynamics will produce much smaller residuals relative to the filter-computed covariance than the mismatched filters. In addition, to prevent the possibility of masking the distinction between the residuals of the filters based on different target dynamics models, the addi-

tion of excessive amounts of pseudonoise to compensate for model inadequacies should be avoided (10:133;14:18).

The computation of the MMAF conditional covariance matrix takes the following form (19:17):

$$P_{mmaf}(t_1^+) = \sum_{k=1}^K p_k(t_1) [P_k(t_1^+) + \hat{y}_k(t_1^+) \hat{y}_k^T(t_1^+)] \quad (2-20)$$

where:

$$\hat{y}_k(t_1^+) = \hat{x}_k(t_1^+) - \hat{x}_{mmaf}(t_1^+)$$

$p_k(t_1)$ = k-th filter's conditional hypothesis probability

$P_k(t_1^+)$ = k-th filter's state error covariance matrix after incorporating the measurement

The values of $P_{mmaf}(t_1^+)$ cannot be computed a priori as can be accomplished for the the case of a linear Kalman filter (as, for instance, each elemental filter) because the values of both $p_k(t_1)$ and $\hat{x}_{mmaf}(t_1^+)$ depend upon the measurements taken through time t_1 .

Finally, to prevent a mismatched filter's $p_k(t_1)$ value from converging to zero, an artificial lower bound is imposed on each of the filter's hypothesis conditional probabilities (10:135;14:18). Without the lower bound, once a conditional probability reaches zero, or essentially reaches zero, it remains zero (or essentially zero) for all time, since it is a function of the conditional probability at the previous time, as seen in Equation (2-15). This effectively impedes the contributions of that filter's estimates for all future times in the MMAF structure and

can degrade the ability of that filter to respond to future changes in the true parameter values. The loss of a filter due to the oversight of not placing a lower bound on its probability value could degrade the MMAP performance if the target dynamics model at a later time matched the model for which the probability was allowed to reach zero, thus preventing it being weighted in the filter structure. The establishment of a lower bound of $p_k(t_i)_{\min} = 0.001$ (8:19) in past efforts is continued in this study.

2.4 Summary

This chapter has presented discussions of both the extended Kalman filter and the concepts underlying multiple model adaptive filtering techniques. The intent of this discussion is to provide a basic understanding of the theory that motivates both the tracking algorithm and the filter model in the ensuing chapters.

III. Truth Model Development

3.1 Introduction

The truth model of any real world filtering and/or control application is the most complete mathematical representation available to the system designer. This truth model becomes the standard with which to evaluate the overall performance of the filter, which is discussed in the next chapter. This chapter describes the characteristics of the truth model as utilized in previous ground-based tracker studies and most recently in (19). However, the modeling of hardware bending and vibration effects is included to represent additional detector perturbations inherent in airborne and other moving-base tracking systems.

For a given tracking scenario, the physical phenomena of target dynamics, FLIR system vibrations, and atmospheric jitter can cause apparent motion between the target and the tracking sensor, where apparent implies some offset from the actual position. Therefore, the location of the centroid of the sensed target image on the FLIR detector plane is a combination of true target motion, a corruption due to optical hardware bending and vibration, and atmospheric jitter caused by infrared wavefront distortion. If x_c represents the azimuthal offset distance of the apparent target centroid from the center of the FLIR plane FOV, then the apparent location of the centroid as detected by the FLIR array is as follows:

$$x_c = x_t + x_a + x_b \quad (3-1)$$

where:

x_t = x_c component due to actual target dynamics

x_a = x_c component due to atmospheric jitter

x_b = x_c component due to bending/vibration

and all components are measured in pixels.

Since the FLIR plane is a two-dimensional array, an equivalent y-direction relationship to Equation (3-1) applies. The truth model that describes this simulation corresponds to a two-state position model augmented by both a six-state atmospheric jitter model and a four-state bending/vibration model. Subsequent sections will describe the dynamics models, followed by the measurement model, target model, and the simulation model which allows computer simulation of the actual tracking scenario.

3.2 Dynamics Model

The twelve-state system dynamics model can be described by the following linear, stochastic differential equation:

$$\dot{x}_T(t) = F_T x_T(t) + B_T u_T(t) + w_T(t) \quad (3-2)$$

where:

F_T = 12 X 12 time-invariant truth model system plant matrix

$x_T(t)$ = 12-dimensional truth model state vector comprised of 2 target position states, 6 atmospheric states, and 4 bending/vibration states

E_T = 12 X 2 time-invariant truth model input distribution matrix

$u_T(t)$ = 2-dimensional truth model deterministic input vector

$w_T(t)$ = 12-dimensional truth model, zero-mean, white Gaussian noise vector process with autocorrelation function:

$$E\{w_T(t)w_T^T(t+\tau)\} = Q_T\delta(\tau) \quad (3-3)$$

The solution to the stochastic differential equation in a discrete-time sense takes the form (Notice that the "T" subscript denoting truth model has been dropped for convenience on the following developments):

$$\underline{x}(t_{i+1}) = \Phi \underline{x}(t_i) + B_d \underline{u}_d(t_i) + \underline{w}_d(t_i) \quad (3-4)$$

where:

$\underline{x}(t_i)$ = 12-dimensional discrete-time state vector

Φ = state transition matrix computed from E_T over the sample period Δt

$$\Delta t = t_{i+1} - t_i$$

$$B_d = \int_{t_i}^{t_{i+1}} \Phi(t_{i+1}, \tau) E_T d\tau$$

$\underline{u}_d(t_i)$ = 2-dimensional discrete-time input vector

$\underline{w}_d(t_i)$ = 12-dimensional discrete-time, zero-mean white Gaussian noise process of covariance:

$$Q_d = \int_{t_i}^{t_{i+1}} \Phi(t_{i+1}, \tau) Q \Phi^T(t_{i+1}, \tau) d\tau \quad (3-5)$$

Recall that the twelve-state discrete-time truth model vector is the result of augmenting a two-state target dynamics vector, $\underline{x}_t(t_1)$, a six-state atmospheric jitter vector, $\underline{x}_a(t_1)$, and a four-state bending and vibration vector, $\underline{x}_b(t_1)$. The partitioning of the truth model dynamics system model solution takes the following form:

$$\begin{bmatrix} \underline{x}_t(t_{i+1}) \\ - \\ \underline{x}_a(t_{i+1}) \\ - \\ \underline{x}_b(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \Phi_t \ 2 \times 2 & | & \underline{0}_{2 \times 6} & | & \underline{0}_{2 \times 4} \\ - & - & - & - & - \\ \underline{0}_{6 \times 2} & | & \Phi_a \ 6 \times 6 & | & \underline{0}_{6 \times 4} \\ - & - & - & - & - \\ \underline{0}_{4 \times 2} & | & \underline{0}_{4 \times 6} & | & \Phi_b \ 4 \times 4 \end{bmatrix} \begin{bmatrix} \underline{x}_t(t_1) \\ - \\ \underline{x}_a(t_1) \\ - \\ \underline{x}_b(t_1) \end{bmatrix} \\
 + \begin{bmatrix} \underline{B}_{dt} \ 2 \times 2 \\ - \\ \underline{0}_{6 \times 2} \\ - \\ \underline{0}_{4 \times 2} \end{bmatrix} \underline{u}_{dt}(t_1) + \begin{bmatrix} \underline{0}_{2 \times 1} \\ - \\ \underline{w}_{da}(t_1) \ 6 \times 1 \\ - \\ \underline{w}_{db}(t_1) \ 4 \times 1 \end{bmatrix} \quad (3-6)$$

where:

\underline{x}_t = 2-dimensional target dynamics state vector

\underline{x}_a = 6-dimensional atmospheric jitter state vector

\underline{x}_b = 4-dimensional bending/vibration state vector

$\underline{u}_{dt}(t_1)$ = 2-dimensional discrete-time deterministic input vector

$\underline{w}_{da}(t_1)$ = 6-dimensional discrete-time white Gaussian noise related to atmospheric states

$\underline{w}_{db}(t_1)$ = 4-dimensional discrete-time white Gaussian noise related to bending/vibration states

Realizing that the block diagonal form of the state transition matrix allows independent evaluation of each particular model, the following subsections develop the propagation relationships for the target, atmospheric jitter, and the bending/vibration models which are ultimately incorporated into Equation (3-6) for the entire system propagation relationship.

3.2.1 Target States. The truth model for target dynamics takes the form of a continuous-time deterministic model representative of the entire range of target trajectories, from highly maneuvering to very benign. Since a FLIR sensor is of primary interest in tracking the target, the elevation, $\alpha(t)$, and azimuth, $\beta(t)$, angles with respect to the center of the FOV on the two-dimensional FLIR image plane become the measurement parameters by which true target location is determined. The target states, therefore, are propagated forward in time with respect to the flat surface approximation α - β plane, where the surface is actually part of a huge sphere.

The α - β plane, as described by (7:36; 18:II-16; 19:26) consists of an array of IR sensing pixels and is perpendicular to the sensor-to-target line-of-sight vector. If the target is far from the FLIR sensor, the FLIR azimuth and elevation angles become proportional to the linear translational coordinates x_t and y_t of the target. Figure 3.1 depicts the relationship between the FLIR angles and the

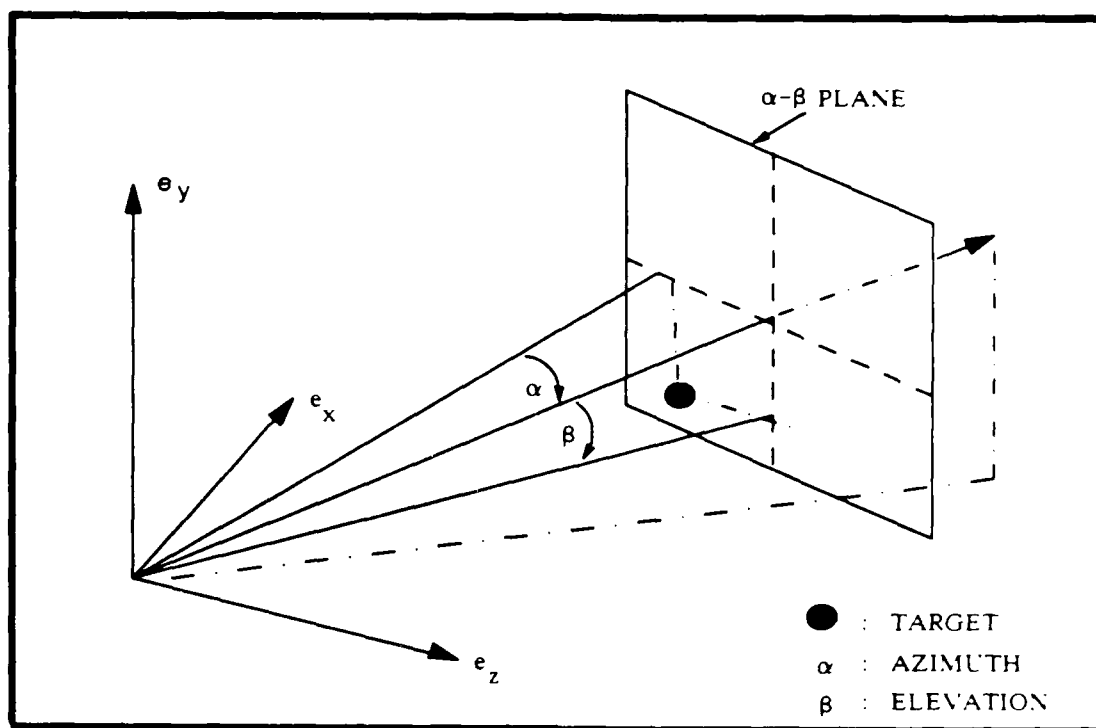


Figure 3.1. The α - β Plane

target centroid. If α and β are measured in micro-radians and x_t and y_t are measured in pixels, then the proportionality constant (k_p) is the angular FOV of a single pixel which has been 20 micro-radians/pixel in previous efforts and developed by Harnly and Jensen (4:33).

If $\dot{\alpha}$ and $\dot{\beta}$ are considered constant over the filter's sample period t seconds in length, then the change in the target position from one sample period to the next can be described in discrete form by:

$$\begin{bmatrix} x_t(t_{i+1}) \\ y_t(t_{i+1}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_t(t_i) \\ y_t(t_i) \end{bmatrix} + \begin{bmatrix} k_p \Delta t & 0 \\ 0 & k_p \Delta t \end{bmatrix} \begin{bmatrix} \dot{\alpha}(t_i) \\ \dot{\beta}(t_i) \end{bmatrix} \quad (3-7)$$

where:

$\dot{\alpha}(t_1) = d\alpha/dt$, measured in micro-radians/second

$\dot{\beta}(t_1) = d\beta/dt$, measured in micro-radians/second

$k_p = 0.05$ pixels/micro-radian

$\Delta t = t_{i+1} - t_i$

Equation (3-7) represents the discrete form solution of the stochastic differential equation for the truth model target states, and thus describes the upper partition of the truth model identified in Section 3.2. This type of model is chosen to fit the state space model form similar to the rest of the dynamics, and it also allows for easy addition of stochastic driving terms, if desired.

3.2.2 Atmospheric States. The atmospheric effects as developed by Mercier (12:73,79) for both the x- and y- directions of the FLIR can be modeled as the output of third order shaping filters driven by white Gaussian noise and represented by the transfer function:

$$\frac{x_a(s)}{w_a(s)} = \frac{K_a \omega_1 \omega_2^2}{(s+\omega_1)(s+\omega_2)^2} \quad (3-8)$$

where:

x_a = atmospheric state shaping filter output

w_a = zero-mean, unit strength, white Gaussian noise

K_a = atmospheric RMS adjustable gain

ω_1 = break frequency, 14.14 rad./sec.

ω_2 = double-pole break frequency, 659.5 rad./sec.

Phase front distortion of the IR radiation waves due to atmospheric disturbances causes translational shifts on the FLIR image plane, called "jitter". This atmospheric jitter perturbation is devoid of directional dependencies; therefore, the modeling of the FLIR plane effects in both the x- and y- directions can be treated separately and identically, and then augmented together to form the atmospheric system description. The effects can be described by the stochastic differential equation:

$$\dot{\underline{x}}_a(t) = \underline{F}_a \underline{x}_a(t) + \underline{w}_a(t) \quad (3-9)$$

where:

$\underline{x}_a(t)$ = six-dimensional atmospheric state vector

\underline{F}_a = atmospheric time-invariant plant matrix

$\underline{w}_a(t)$ = six-dimensional, zero-mean, white Gaussian noise process of strength \underline{Q}_a

When the state vector and the plant matrix are expressed in Jordan canonical form (12:212), the \underline{Q}_a matrix appears as (19:31):

$$\underline{Q}_a = \begin{bmatrix} Q_1^2 & Q_1 Q_2 & Q_1 Q_3 & 0 & 0 & 0 \\ Q_1 Q_2 & Q_2^2 & Q_2 Q_3 & 0 & 0 & 0 \\ Q_1 Q_3 & Q_2 Q_3 & Q_3^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_1^2 & Q_1 Q_2 & Q_1 Q_3 \\ 0 & 0 & 0 & Q_1 Q_2 & Q_2^2 & Q_2 Q_3 \\ 0 & 0 & 0 & Q_1 Q_3 & Q_2 Q_3 & Q_3^2 \end{bmatrix} \quad (3-10)$$

where:

$$Q_1 = (K\omega_1\omega_2^2)/(\omega_1-\omega_2)^2$$

$$Q_2 = -Q_1$$

$$Q_3 = (K\omega_1\omega_2^2)/(\omega_1-\omega_2)$$

The truth model's center partition in Equation (3-6) is formed by representing Equation (3-9) in discrete form. The statistics of the discrete-time noise covariance can be described by the following integral:

$$Q_{da} = \int_0^t \Phi_a(\tau) Q_a \Phi_a^T(\tau) d\tau \quad (3-11)$$

where:

Q_{da} = 6 X 6 atmospheric discrete-time noise covariance matrix

$\Phi_a(\tau)$ = 6 X 6 atmospheric state transition matrix associated with \bar{E}_a of Eq. (3-9)

and the 6 X 6 atmospheric state transition matrix forms the center partition of the 12 X 12 truth model state transition matrix. The atmospheric state transition matrix has been developed and can be shown to be of the form (19:32):

$$\Phi_a(\Delta t) = \begin{bmatrix} e^{-\omega_1 \Delta t} & 0 & 0 & 0 & 0 & 0 \\ 0 & e^{-\omega_2 \Delta t} & -\omega_2 \Delta t & 0 & 0 & 0 \\ 0 & 0 & e^{-\omega_2 \Delta t} & 0 & 0 & 0 \\ 0 & 0 & 0 & e^{-\omega_1 \Delta t} & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{-\omega_2 \Delta t} & -\omega_2 \Delta t \\ 0 & 0 & 0 & 0 & 0 & e^{-\omega_2 \Delta t} \end{bmatrix} \quad (3-12)$$

3.2.3 Mechanical States. The modeling of the mechanical bending/vibration effects is based on a report by R & D Associates (15) that was prepared for AFWL concerning dynamic modelling for space based structures. Figure 3.2 depicts the response spectrum for the x-direction line-of-sight for a particular satellite structure and the approximated curve to implement a reduced-order model that is representative of this type of response. The effects due to the bending and vibration of the mechanical structure of the optical equipment are developed in Appendix A. Since the effects are similar in both the x- and y-directions of the FLIR, both directions can be modeled identically by means of a second order shaping filter driven by white Gaussian noise and represented by the transfer function:

$$\frac{x_b(s)}{w_b(s)} = \frac{K_b \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} \quad (3-13)$$

where:

x_b = mechanical state shaping filter output

w_b = zero-mean, unit strength, white Gaussian noise with an autocorrelation:

$$E(w_b(t)w_b(t-\tau)) = Q_b \delta(t-\tau); Q_b = 1 \quad (3-14)$$

K_b = gain adjusted to obtain the desired root mean square (RMS) bending/vibration output

ζ = damping coefficient, = .15

ω_n = natural mechanical frequency, = π rad./sec.

A closer approximation to the low frequency spectra of Figure 3.2 can be obtained by representing each direction as a sixth order system versus a second order; however, twelve

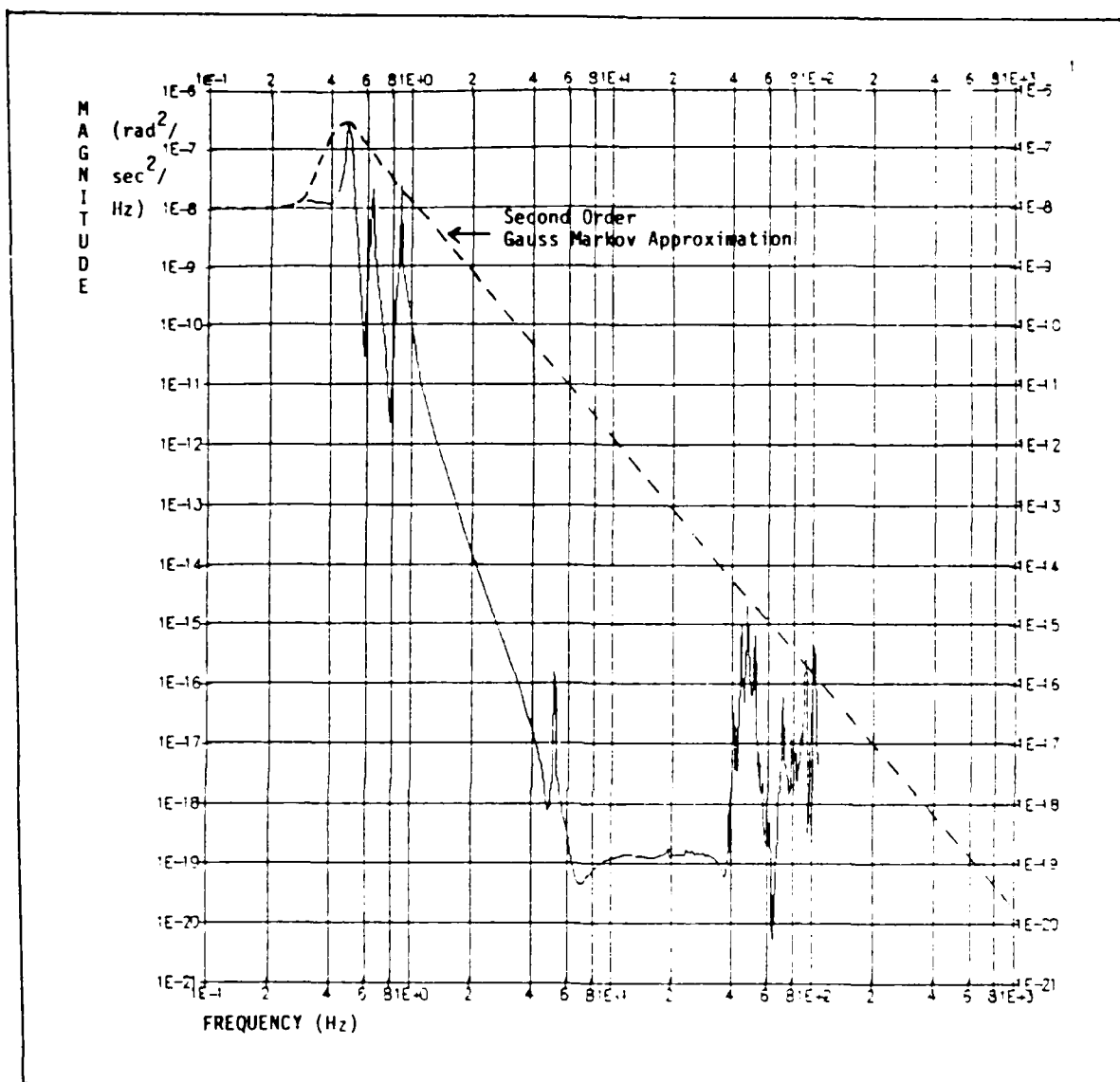


Figure 2.2. Response Spectrum for LOS x-direction.

states versus four for the two axes was deemed as beyond the point of diminishing returns for a first-cut analysis, and the modeling of the fundamental frequency effect of bending was determined to be sufficient enough to describe the effect. The modeling of the effects in both the x- and y-

directions can be treated independently and augmented together to form the mechanical system description. These effects can be described by the stochastic differential equation of the following form:

$$\dot{x}_b(t) = F_b x_b(t) + G_b w_b(t) \quad (3-14)$$

where:

$x_b(t)$ = four-dimensional mechanical state vector

F_b = time invariant mechanical plant matrix

$w_b(t)$ = two-dimensional, zero-mean, white Gaussian noise process of strength $Q_b = I$

G_b = 4 X 2 noise distribution matrix

As developed in Appendix A, the Q_b matrix is shown as:

$$Q_b = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3-15)$$

The truth model's lower partition in Equation (3-6) is formed by representing Equation (3-14) in discrete form. The value of the discrete-time mechanical noise covariance can be described by the following integral (9:171):

$$Q_{bd} = \int_0^{\Delta t} \Phi_b(\tau) G_b Q_b G_b^T \Phi_b^T(\tau) d\tau \quad (3-16)$$

where:

Q_{bd} = 4 X 4 discrete-time mechanical noise covariance matrix

Φ_b = 4 X 4 mechanical vibration state transition matrix associated with F_b of Eq. (3-14)

Δt = sample period, $t_{i+1} - t_i$

and the 4 X 4 mechanical vibration state transition matrix forms the lower partition of the 12 X 12 truth model state transition matrix. The mechanical state transition matrix as developed in Appendix A results in a 4 X 4 block diagonal matrix of two identical 2 X 2 blocks with the following form:

$$\Phi_b = \begin{bmatrix} \phi_{b1} & \phi_{b2} & 0 & 0 \\ \phi_{b3} & \phi_{b4} & 0 & 0 \\ 0 & 0 & \phi_{b1} & \phi_{b2} \\ 0 & 0 & \phi_{b3} & \phi_{b4} \end{bmatrix} \quad (3-17)$$

where:

$$\phi_{b1} = \exp(-\sigma_b \Delta t) \{ \cos(\omega_b \Delta t) + (\sigma_b / \omega_b) \sin(\omega_b \Delta t) \}$$

$$\phi_{b2} = \exp(-\sigma_b \Delta t) \{ (1 / \omega_b) \sin(\omega_b \Delta t) \}$$

$$\phi_{b3} = \exp(-\sigma_b \Delta t) \{ -(1 + (\sigma_b / \omega_b)^2) \sin(\omega_b \Delta t) \}$$

$$\phi_{b4} = \exp(-\sigma_b \Delta t) \{ \cos(\omega_b \Delta t) - (\sigma_b / \omega_b) \sin(\omega_b \Delta t) \}$$

σ_b = real part to the characteristic equation of Equation (3-13)

ω_b = imaginary part to the characteristic equation of Equation (3-13)

3.3 Measurement Model

The measurements accessible to the tracking algorithm represent the average intensity of target and background radiation incident on the FLIR array of detector elements. The measured IR image, or intensity function, consists of a collection of information from the target IR radiation, background noises, and FLIR sensor noises. Previous research (18:11-20) has determined that single target hotspot patterns could be modeled as a bivariate Gaussian

function with constant-intensity contours when describing distant targets. This bivariate Gaussian intensity function can be described by :

$$I(x, y, x_{\text{peak}}(t), y_{\text{peak}}(t)) = I_{\text{max}} \exp \left\{ -0.5 \left[(x - x_{\text{peak}})(y - y_{\text{peak}}) \right] \right. \\ \left. [P_{\alpha\beta}]^{-1} \left[(x - x_{\text{peak}})(y - y_{\text{peak}}) \right]^T \right\} \quad (3-18)$$

where:

I_{max} = maximum intensity of the hotspot

$(x_{\text{peak}}, y_{\text{peak}})$ = coordinates of the centroid of the apparent target intensity profile on the image plane, measured in pixels

x, y = spatial coordinates calculated relative to the center of the tracker FOV.

$P_{\alpha\beta}$ = 2 X 2 matrix whose eigenvalues (σ_v^2 and σ_{pv}^2) define the dispersion of the elliptical constant intensity contours in the α - β frame. The eigenvectors of $P_{\alpha\beta}$ define the principal axes of the bivariate distribution.

Although single hotspot IR sources can be represented by bivariate Gaussian distributions, targets with multiple hotspots cannot. Figure 3.3 shows the FLIR plane intensity profile for a three hotspot target (14:35). The centroid of the apparent target intensity profile cannot be represented as a Gaussian distribution even though the individual hotspots are modeled as Gaussian with elliptical constant-intensity contours. Furthermore, the intensity measurement provided by each pixel corresponds to the average intensity on that pixel resulting from the effects of the target intensity function, the spatially correlated background noise, and the FLIR sensor noise.

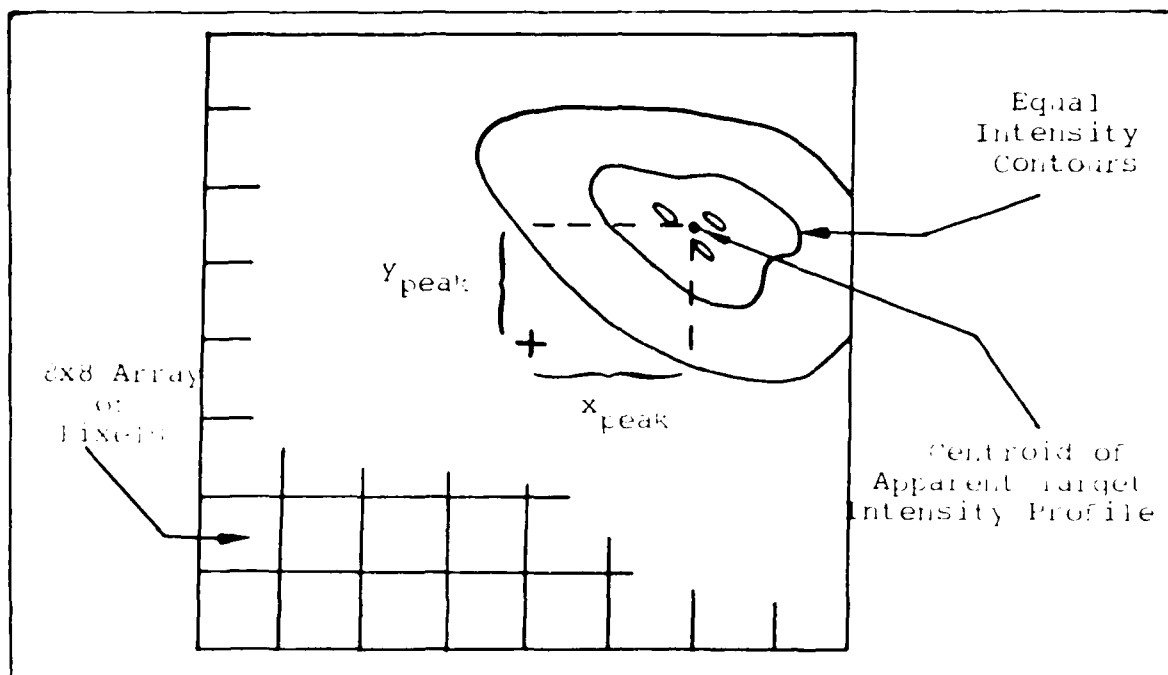


Figure 3.3. FLIR Image Plane Intensity Function for a Three Hotspot Target

The measured output for the pixel in the j -th row and k -th column at sample time t_1 is described by (13:33,34):

$$z_{jk}(t_1) = \sum_{m=1}^M \{ (1/A_p) \int_{\text{pixel } jk} I_m(x, y, x_{peakm}(t_1), y_{peakm}(t_1)) dx dy \} + n_{jk}(t_1) + b_{jk}(t_1) \quad (3-19)$$

where:

$z_{jk}(t_1)$ = output of pixel jk

A_p = area of one pixel

M = total number of hotspots

$I_m(\cdot)$ = intensity function due to m -th hotspot

(x, y) = coordinates of a point within pixel jk

(x_{peakm}, y_{peakm}) = coordinates of maximum intensity point of m -th hotspot

$n_{jk}(t_1)$ = effect of sensor errors on pixel jk

$b_{jk}(t_1)$ = effect of background noise on pixel jk

Temporally and spatially uncorrelated thermal noise and dark current in the IR sensitive pixels comprise the sensor error, $n_{jk}(t_1)$ (19:35); whereas the background noise, $b_{jk}(t_1)$, can be represented as a spatially correlated noise. This spatially correlated noise is modeled as an exponentially decaying correlation pattern with radial symmetry and characterized by a correlation distance of approximately two pixels (4:37,40). This is simulated by maintaining non-zero correlation coefficients between pixels separated from each other by two pixels or less in all directions (19:35).

3.4 Target Model

A three hotspot planform, with a coordinate frame that is discussed in Section 3.5.1, is depicted in Figure 3.4. It is considered to represent the target of interest for this analysis. The target's angle of attack and sideslip angle are assumed to be zero in order to maintain the alignment of the semi-major axes of each of the three elliptical IR constant-intensity contours with the target's velocity vector. This enables a simplistic simulation of the space geometry without substantially degrading tracker performance analysis (19:38).

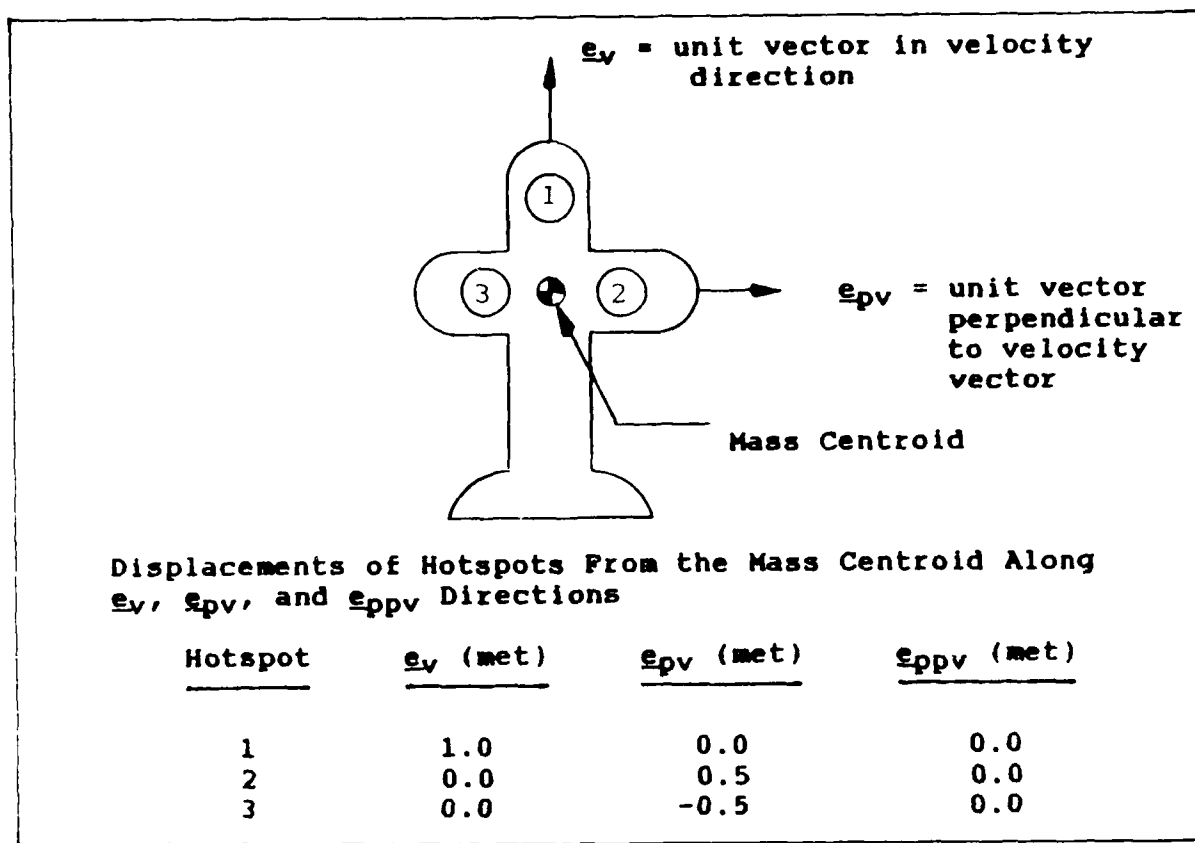


Figure 3.4. Target Hotspot Distribution

3.4.1 Target Trajectories. The target trajectories available in this study include the four trajectories used by Tobin (19:38,41), and two additional target scenarios as shown in Figure 3.5. Each is described as follows:

Trajectory One. This trajectory is a constant velocity, straight and level flight with a simulation time of five seconds. The inertial velocity remains in a plane parallel to the \underline{e}_x - \underline{e}_z plane. Note that it is not constant velocity when projected onto the FLIR image plane because the tracker is a rotating frame.

Trajectory Two. This trajectory involves a constant 10-g or 20-g pull-up maneuver that is represented by a step change in the acceleration at $t=2.0$ seconds and lasts for the remainder of the 5 second simulation. The target's turn is portrayed by a constant-magnitude angular rate vector oriented parallel to the negative e_z axis.

Trajectory Three. This trajectory is similar to Trajectory Two with the exception that at $t=3.5$ seconds, the constant-g pull-up maneuver abruptly terminates and the pitch rate is set to zero for the remainder of the 6.67 second simulation. This longer simulation interval allows the tracker time to settle down and corresponds to 200 sample periods of duration.

Trajectory Four. This trajectory is similar to Trajectory Two with the exception that the angular rate vector is parallel to the e_y axis, which simulates the target turning towards the tracker.

Trajectory Five. This is a Trajectory Two rotated at an angle of 45° with respect to the FLIR plane. The intent of this scenario is to evaluate how well the tracking algorithm performs against a maneuvering target for which the tracking algorithm is not tuned (the appropriate direction of elongated fields-of-view, to be discussed in Section 5.4, would be neither the x- nor the y- direction in the FLIR image plane).

Trajectory Six. This trajectory is identical to Trajectory Two with the exception that the acceleration to

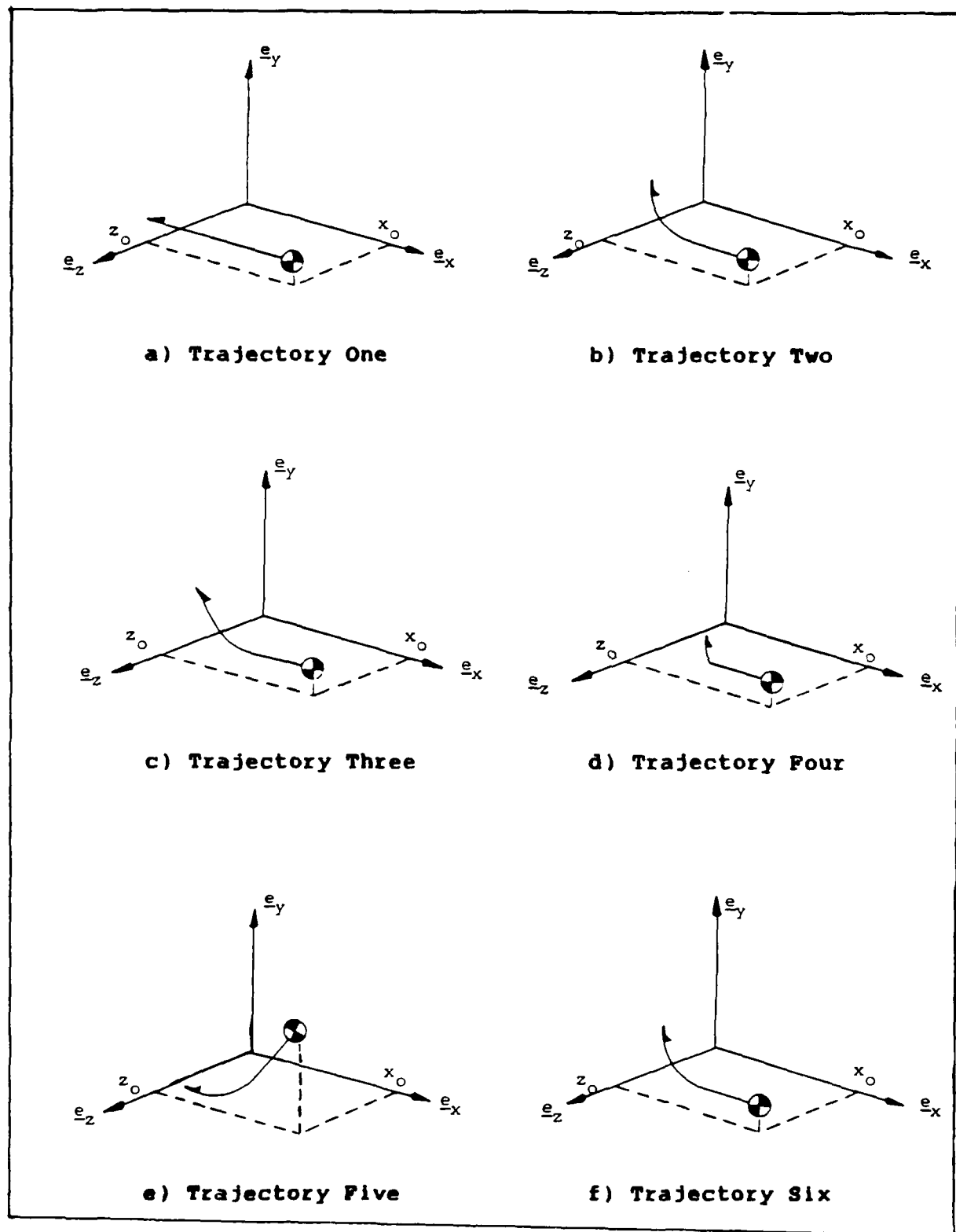


Figure 3.5. Target Trajectories

start the pull-up maneuver is allowed to develop as a positively increasing ramp rather than as a step change. Previous studies had modeled the accelerated turns with a step change in acceleration, a rather harsh maneuver for the tracking algorithm to maintain lock and rather unrealistic for physical targets to perform.

3.5 Simulation Space Model

The simulation space model allows the simulation of the FLIR tracker operation on a digital computer. This model allows for the realistic target trajectory propagation in three dimensional space and provides a mathematical means of describing the target's image and velocity vector with respect to the FLIR image plane (19:36). Both the FLIR plane velocity projection and the FLIR plane target image projection are discussed following the description of the various coordinate frames.

3.5.1 Coordinate Frames. The coordinate frames used for the FLIR tracker digital simulation include the target frame, the inertial frame, the $\alpha - \beta - r$ frame and the $\alpha - \beta$ (FLIR image) frame. The origin of each frame and the orientation of the reference axes are as follows:

Target Frame:

Origin: target center of mass

Axes: \underline{e}_v - along velocity vector

\underline{e}_{pv} - out right side of target,
perpendicular to \underline{e}_v

\underline{e}_{ppv} - vector completing right hand coordinate set, perpendicular to both \underline{e}_v and \underline{e}_{pv}

Inertial Frame:

Origin: location of FLIR sensor

Axes: \underline{e}_x - due north, tangent to earth's surface, defines zero azimuth

\underline{e}_y - inertial 'up' with respect to a flat earth approximation

\underline{e}_z - vector completing right hand coordinate set, defines 90° azimuth

Note: Azimuth (α) is measured eastward from \underline{e}_x . Elevation (β) is measured from the plane defined by \underline{e}_x and \underline{e}_z .

$\alpha - \beta - r$ Frame:

Origin: target center of mass

Axes: \underline{e}_r - coincident with the true sensor-to-target LOS vector

$\underline{e}_\alpha, \underline{e}_\beta$ - define a plane perpendicular to \underline{e}_r , rotated from the inertial \underline{e}_x and \underline{e}_y by the azimuth and elevation angles

$\alpha - \beta$ (FLIR Image) Plane:

The FLIR image plane by which the sensor measurements are taken comprise this plane. The azimuth and elevation angles can be considered the linear translational coordinates x and y . Observing the FLIR plane along the LOS vector, x is chosen to be positive to the right and y is positive down. This choice of coordinates maintains a right-handed coordinate set with the target's range measured positive away from the sensor.

Both the inertial and $\alpha - \beta$ frames are depicted in Figure 3.1, whereas the target frame is illustrated in Figure 3.3.

3.5.2 FLIR Plane Velocity Projection. The target inertial velocities are transformed into the deterministic azimuth velocity, $\dot{\alpha}(t)$, and elevation velocity, $\dot{\beta}(t)$, in the truth model differential equation, Equation (3-2), by projecting the inertial velocities onto the FLIR image plane based on the geometry of Figure 3.6 which was utilized by Loving (8:27,29). From Figure 3.6(b), α can be derived by the relationship:

$$\alpha(t) = \tan^{-1} [z(t)/x(t)] \quad (3-20)$$

and $\beta(t)$ can be derived from Figure 3.6(c) by:

$$\beta(t) = \tan^{-1} [y(t)/r_h(t)] \quad (3-21)$$

where:

$$r_h(t) = \text{horizontal range; } = (x^2 + z^2)^{1/2}$$

However, the variables of interest for velocity projections are $\dot{\alpha}(t)$ and $\dot{\beta}(t)$. Therefore, taking the time derivative of Equations (3-20) and (3-21) yields:

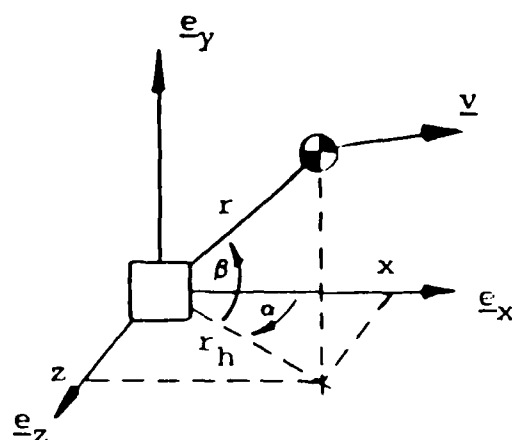
$$\dot{\alpha}(t) = \frac{x(t)v_z(t) - z(t)v_x(t)}{x^2(t) + z^2(t)} \quad (3-22)$$

and

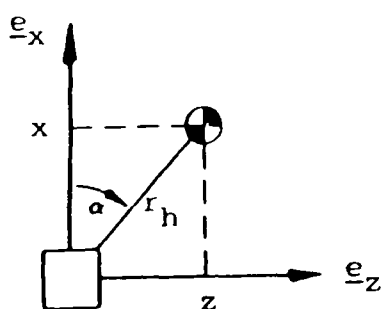
$$\dot{\beta}(t) = \frac{r_h(t)v_y(t) - y(t)\dot{r}_h(t)}{r_h^2(t)} \quad (3-23)$$

where:

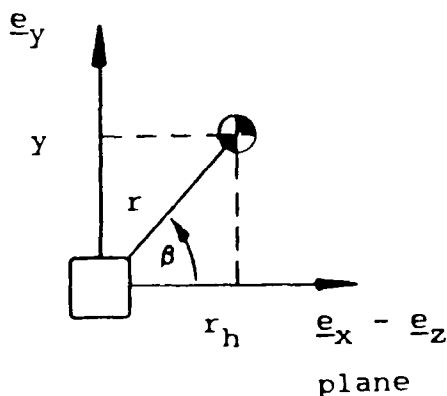
v_x, v_y, v_z = target velocity components in the inertial e_x, e_y , and e_z directions, respectively



(a). Target/Inertial Frame Geometry



(b). Azimuth Geometry



(c). Elevation Geometry

KEY



: Sensor



: Target

$\underline{e}_x, \underline{e}_y, \underline{e}_z$: inertial axes

r : sensor-to-target range; $= (x^2 + y^2 + z^2)^{1/2}$

r_h : horizontal range; $= (x^2 + z^2)^{1/2}$

\underline{v} : inertial velocity of target

α : azimuth angle

β : elevation angle

Figure 3.6. Geometry to Project Target Inertial Velocity onto the FLIR Image Plane

and

$$\dot{r}_h(t) = \frac{x(t)v_x(t) + z(t)v_z(t)}{r_h(t)} \quad (3-24)$$

Equations (3-22) and (3-23) define the deterministic input vector in the truth model difference equation, shown in Equation (3-6), where $\underline{u}_{dt}(t_i) = [\dot{\alpha}(t_i), \dot{\beta}(t_i)]^T$.

3.5.3 FLIR Plane Target Image Projection. The projection of the target's hotspots onto the FLIR image plane corresponds to the the detection of the target by the array of pixels. Although the hotspots are assumed to lie in the plane formed by the wings of the target, the orientation and location of the hotspots on the FLIR image plane changes as the target rotates and translates with respect to the IR sensor. Figure 3.7 illustrates the geometry of the target with inertial velocity vector assumed

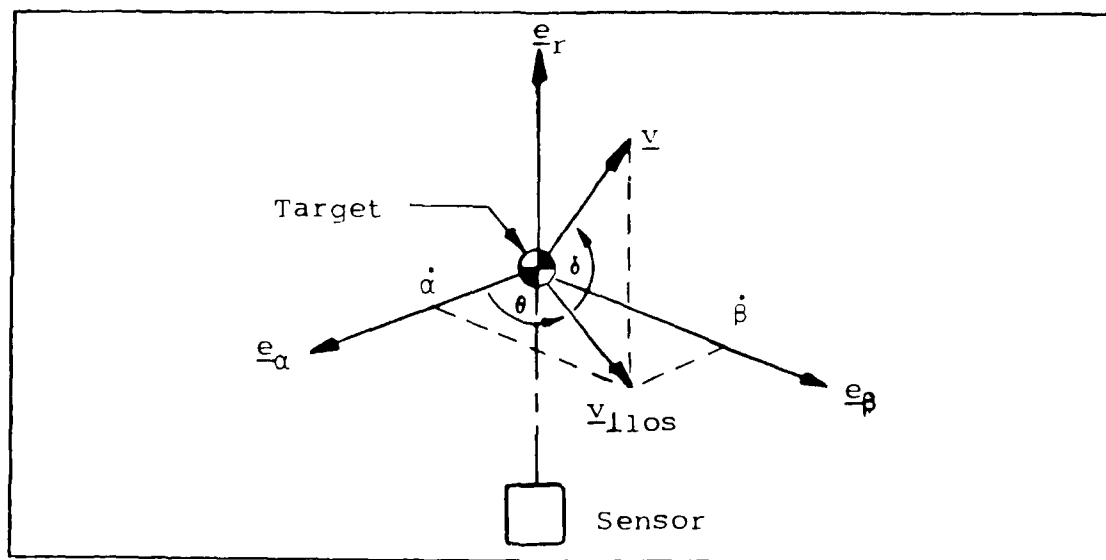


Figure 3.7. Target Image Projection Geometry

to project through the nose of the target, the α - β plane perpendicular to the true LOS from the tracker to the target, and the origin of the α - β plane coinciding with the detected target center of mass. The image of the target as perceived by the IR sensor changes as the target changes angular orientation relative to the tracker or moves farther away from or closer to the FLIR. The current image size of the target with respect to the largest planform image, or reference image, is given by the following (12:II-24):

$$\sigma_{pv} = \sigma_{pvo} (r_0/r) \quad (3-25)$$

and

$$\begin{aligned} \sigma_v &= (r_0/r) [\sigma_{pvo} + (\sigma_{vo} - \sigma_{pvo}) \cos \delta] \\ &= \sigma_{pv} \{1 + [(v_{\perp los})/v] [AR - 1]\} \end{aligned} \quad (3-26)$$

where:

$\sigma_{vo}, \sigma_{pvo}$ = target hotspot dispersions along \underline{e}_v and \underline{e}_{pv} in the target frame of the reference image

σ_v, σ_{pv} = target image's current dispersions

r_0 = sensor-to-target range of reference image

r = current sensor-to-target range

\underline{v} = target inertial velocity vector

v = magnitude of \underline{v}

$\underline{v}_{\perp los}$ = component of \underline{v} perpendicular to the LOS vector (projected onto the α - β plane)

$v_{\perp los}$ = magnitude of $\underline{v}_{\perp los} = [\dot{\alpha}^2 + \dot{\beta}^2]^{1/2}$

δ = angle between \underline{v} and the α - β plane

AR = aspect ratio of reference image, σ_{vo}/σ_{pvo}

Since the ultimate goal is to define the target image in terms of the FLIR plane coordinates, a transformation from the target frame coordinates to the α - β coordinates can be expressed as:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{\alpha\beta} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}_{\text{target frame}} = \underline{T} \underline{x} \quad (3-27)$$

where:

θ = angle between \underline{e}_α and $\underline{v}_{\perp\text{LOS}}$ (see Figure 3.7)

The dispersion matrix can be determined from the following:

$$\underline{P}_{\alpha\beta} = \underline{T} \underline{P} \underline{T}^T \quad (3-28)$$

where the dispersion matrix is utilized in the measurement model in Equation (3-18).

3.6 Summary

This chapter has introduced several models which depict the real world target tracking environment. The processes that have been modeled include: target dynamics, atmospheric jitter, mechanical bending/vibration, and background and FLIR noise. Deterministic target trajectories were described to provide the baseline for testing the tracker against realistic tracking scenarios. Finally, a simulation space was presented in order to allow the simulation of the tracking scenario on a digital computer.

IV. Filter Models

4.1 Introduction

This chapter describes the filter models employed in the MMAP structure discussed in Chapter II. Section 4.2 discusses the reduced-order dynamics models: the Gauss-Markov acceleration model and the constant turn-rate dynamics model, and Section 4.3 discusses the enhanced-correlator/linear-measurement model.

4.2 Dynamics Models

Two distinct dynamics models are presented for the purpose of comparative analyses in the MMAP. Both models represent a reduced-order, eight-state vector to estimate the target's position, velocity, and acceleration, and the atmospheric jitter in two orthogonal directions (x,y) on the FLIR image plane. The filter state vector for each of the two models is represented as:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \\ v_x \\ v_y \\ a_x \\ a_y \\ x_a \\ y_a \end{bmatrix} \quad (4-1)$$

where:

x_t, y_t = target position, x- and y-direction

v_x, v_y = target velocity, x- and y-direction

a_x, a_y = target acceleration, x- and y-direction

x_a, y_a = atmospheric jitter, x- and y-direction

Note that the atmospheric jitter is represented by two filter states rather than six and that the filter neglects the modelling of the bending/vibration states as presented in the truth model of Chapter III. The filter's atmospheric jitter model captures the primary characteristics of the jitter's power spectral density while at the same time neglecting the high frequency effects. The omission of modeling the bending/vibration states within the filters is be presented as part of the analysis on filter performance in Chapter VI.

Although the state vectors that describe the two dynamics models to be investigated are the same, the description of the target's acceleration process is uniquely defined for each model. The linear filter of Section 4.2.1 models the target's inertial acceleration as a zero-mean, first order Gauss-Markov process, while the nonlinear filter of Section 4.2.2 models the target's trajectory as described by a series of concatenated constant turn-rate segments.

4.2.1 Gauss-Markov Acceleration Model. This dynamics model, a standard airborne target model not originating in these references, assumes that the target's inertial

acceleration can be described as the output of a first-order lag driven by white, Gaussian noise. Developed by Millner (13:50,51) and Kozemchak (7:75,80) for the FLIR tracker, it is well described by the time invariant, linear stochastic differential equation:

$$\dot{\underline{x}}_f(t) = \underline{F}_f \underline{x}_f(t) + \underline{G}_f \underline{w}_f(t) \quad (4-2)$$

where:

$\underline{x}_f(t)$ = 8-dimensional filter state vector

$\underline{w}_f(t)$ = 4-dimensional, zero-mean, white Gaussian vector noise process of strength \underline{Q}_f .

$$\underline{F}_f = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1/\tau_x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/\tau_y & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_a & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_a \end{bmatrix} \quad (4-3)$$

$$\underline{Q}_f = \begin{bmatrix} 2\sigma_x^2/\tau_x & 0 & 0 & 0 \\ 0 & 2\sigma_y^2/\tau_y & 0 & 0 \\ 0 & 0 & 2\sigma_a^2/\tau_a & 0 \\ 0 & 0 & 0 & 2\sigma_a^2/\tau_a \end{bmatrix} \quad (4-4)$$

$$\underline{Q}_f = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4-5)$$

where:

τ_x, τ_y = correlation time for the x- and y- target acceleration processes

τ_a = correlation time for atmospheric jitter processes

σ_x^2, σ_y^2 = variance and mean-squared value for the x- and y- target acceleration

σ_a^2 = variance and mean-squared value for the atmospheric jitter

The filter state estimate and error covariance matrix are propagated forward in a discrete-time sense by the following set of equations (9:171,172):

$$\hat{\underline{x}}_f(t_{i+1}^-) = \underline{\Phi}_f(t) \hat{\underline{x}}_f(t_i^+) \quad (4-6)$$

$$\underline{P}_f(t_{i+1}^-) = \underline{\Phi}_f(\Delta t) \underline{P}_f(t_i^+) \underline{\Phi}_f^T(\Delta t) + \underline{Q}_{fd} \quad (4-7)$$

where:

$\hat{\underline{x}}_f(t_i)$ = filter's estimate of the state vector

$\underline{P}_f(t_i)$ = filter's error covariance matrix

(t_1^-) = instant before the measurement is incorporated at time t_1

(t_1^+) = instant after the measurement is incorporated at time t_1

$\Phi_f(\Delta t)$ = time invariant state transition matrix associated with F_f for propagation over the sample period t :

$$\Delta t = t_{i+1} - t_i$$

$$Q_{fd} = \int_{t_i}^{t_{i+1}} \Phi_f(t_{i+1}, \tau) Q_f Q_f^T \Phi_f^T(t_{i+1}, \tau) d\tau$$

The $\Phi_f(\Delta t)$ and Q_{fd} matrices have been determined to be of the following form (14:47,48):

$$\Phi_f(\Delta t) = \begin{bmatrix} 1 & 0 & \Delta t & 0 & \phi_{15} & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & \phi_{26} & 0 & 0 \\ 0 & 0 & 1 & 0 & \phi_{35} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \phi_{46} & 0 & 0 \\ 0 & 0 & 0 & 0 & \phi_{55} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \phi_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \phi_{77} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \phi_{88} \end{bmatrix} \quad (4-8)$$

where:

$$\phi_{15} = \tau_x(\Delta t - \tau_x(1 - \exp(-\Delta t/\tau_x)))$$

$$\phi_{26} = \tau_y(\Delta t - \tau_y(1 - \exp(-\Delta t/\tau_y)))$$

$$\phi_{35} = \tau_x(1 - \exp(-\Delta t/\tau_x))$$

$$\phi_{46} = \tau_y(1 - \exp(-\Delta t/\tau_y))$$

$$\phi_{55} = \exp(-\Delta t/\tau_x)$$

$$\phi_{66} = \exp(-\Delta t/\tau_y)$$

$$\phi_{77} = \phi_{88} = \exp(-\Delta t/\tau_a)$$

and

$$Q_{fd} = \begin{bmatrix} Q_{11} & 0 & Q_{13} & 0 & Q_{15} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & Q_{24} & 0 & Q_{26} & 0 & 0 \\ Q_{13} & 0 & Q_{33} & 0 & Q_{35} & 0 & 0 & 0 \\ 0 & Q_{24} & 0 & Q_{44} & 0 & Q_{46} & 0 & 0 \\ Q_{15} & 0 & Q_{35} & 0 & Q_{55} & 0 & 0 & 0 \\ 0 & Q_{26} & 0 & Q_{46} & 0 & Q_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Q_{77} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{88} \end{bmatrix} \quad (4-9)$$

where:

$$Q_{11} = \sigma_x^2 \{ (2/3) \tau_x \Delta t^3 - 2(\tau_x \Delta t)^2 - 4\tau_x^3 (\Delta t) \exp(-\Delta t/\tau_x) + 2\tau_x^3 \Delta t - \tau_x^4 \exp(-2\Delta t/\tau_x) + \tau_x^4 \}$$

$$Q_{13} = \sigma_x^2 \{ \tau_x \Delta t^2 + 2\tau_x^2 (\Delta t) \exp(-\Delta t/\tau_x) + \tau_x^3 - 2\tau_x^3 \exp(-\Delta t/\tau_x) - 2\tau_x^2 \Delta t + \tau_x^3 \exp(-2\Delta t/\tau_x) \}$$

$$Q_{15} = \sigma_x^2 \{ -2\tau_x (\Delta t) \exp(-\Delta t/\tau_x) + \tau_x^2 - \tau_x^2 \exp(-2\Delta t/\tau_x) \}$$

$$Q_{22} = \sigma_y^2 \{ (2/3) \tau_y \Delta t^3 - 2(\tau_y \Delta t)^2 - 4\tau_y^3 (\Delta t) \exp(-\Delta t/\tau_y) + 2\tau_y^3 \Delta t - \tau_y^4 \exp(-2\Delta t/\tau_y) + \tau_y^4 \}$$

$$Q_{24} = \sigma_y^2 \{ \tau_y \Delta t^2 + 2\tau_y^2 (\Delta t) \exp(-\Delta t/\tau_y) + \tau_y^3 - 2\tau_y^3 \exp(-\Delta t/\tau_y) - 2\tau_y^2 \Delta t + \tau_y^3 \exp(-2\Delta t/\tau_y) \}$$

$$Q_{26} = \sigma_y^2 \{-2\tau_y(\Delta t)\exp(-\Delta t/\tau_y) + \tau_y^2 - \tau_y^2\exp(-2\Delta t/\tau_y)\}$$

$$Q_{33} = \sigma_x^2 \{2x(\Delta t) - 3\tau_x^2 + 4\tau_x^2\exp(-\Delta t/\tau_x) - \tau_x^2\exp(-2\Delta t/\tau_x)\}$$

$$Q_{35} = \sigma_x^2 \{\tau_x - 2\tau_x\exp(-\Delta t/\tau_x) + \tau_x\exp(-2\Delta t/\tau_x)\}$$

$$Q_{44} = \sigma_y^2 \{2\tau_y\Delta t - 3\tau_y^2 + 4\tau_y^2\exp(-\Delta t/\tau_y) - \tau_y^2\exp(-2\Delta t/\tau_y)\}$$

$$Q_{46} = \sigma_y^2 \{\tau_y - 2\tau_y\exp(-\Delta t/\tau_y) + \tau_y\exp(-2\Delta t/\tau_y)\}$$

$$Q_{55} = \sigma_x^2 \{1 - \exp(-2\Delta t/\tau_x)\}$$

$$Q_{66} = \sigma_y^2 \{1 - \exp(-2\Delta t/\tau_y)\}$$

$$Q_{77} = \sigma_a^2 \{1 - \exp(-2\Delta t/\tau_a)\}$$

$$Q_{88} = Q_{77}$$

The Multiple Model Adaptive Filter estimates are formed via a probabilistically weighted averaging technique.

Following each propagation cycle, the elemental filter estimates, $\hat{x}_1(t_{i+1}^-) = \hat{x}_t(t_{i+1}^-)$ and $\hat{x}_2(t_{i+1}^-) = \hat{y}_t(t_{i+1}^-)$ form the MMAP estimates, $\hat{x}_1(t_{i+1}^-)_{\text{mmaf}}$ and $\hat{x}_2(t_{i+1}^-)_{\text{mmaf}}$. Analogously, the estimates $\hat{x}_{\text{mmaf}}(t_i^+)$ could also be used. These estimates can be considered as the control signals that, if driven to zero over the next sample period, would drive the FLIR optical centerline at the target.

4.2.2 Constant Turn-Rate Dynamics Model. The constant turn-rate dynamics model is an alternative to the linear, first-order Gauss Markov acceleration process, and it is described by a series of concatenated constant turn-rate

segments (19:55,58). Since this model results in a nonlinear dynamics model, the extended Kalman filter propagation equations described in Chapter II are utilized. The constant turn-rate target acceleration process is modelled by the following first-order, nonlinear, stochastic differential equation:

$$\dot{\underline{a}}(t) = -\omega^2 \underline{v}(t) + \underline{w}(t) \quad (4-10)$$

where:

ω = angular velocity of the constant turn

$$= \frac{\underline{v}(t) \times \underline{a}(t)}{|\underline{v}(t)|^2} \quad (4-11)$$

$\underline{a}(t)$ = 2-dimensional target acceleration vector;
coordinatized in the FLIR image plane

$\underline{v}(t)$ = 2-dimensional target velocity vector;
coordinatized in the FLIR image plane

$\underline{w}(t)$ = 2-dimensional zero-mean, white Gaussian
vector noise process

This model results in a nonlinear system state equation of the form of Equation (2-1) when utilizing the state variables defined in Equation (4-1). The components of $\underline{f}(\underline{x}(t), t)$ are identical to those of Equation (4-3) with exception to the f_5 and f_6 components. These two components are functions of the nonlinear vector function defined in Equations (4-10) and (4-11). The noise distribution matrix, \underline{G}_f , is identical to Equation (4-4) and the noise strength matrix, \underline{Q}_f , can be determined via an iterative tuning process depending on the simulation scenario parameters.

The state estimate and error covariance matrix are propagated between sample periods by employing the extended Kalman filter equations defined in Section 2.1. The sensitivity matrix, \underline{F} , of Equation (2-11) is described by the partial differential matrix equation:

$$\underline{F} = \frac{\partial \underline{f}}{\partial \underline{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & F_{53} & F_{54} & F_{55} & F_{56} & 0 & 0 \\ 0 & 0 & F_{63} & F_{64} & F_{65} & F_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_a & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_a \end{bmatrix} \quad (4-12)$$

where:

$$F_{53} = \frac{A}{B^2} \left(-3x_3x_6 + x_4x_5 + \frac{4x_3^2A}{B} \right)$$

$$F_{54} = \frac{A}{B^2} \left(2x_3x_5 + \frac{4x_3x_4A}{B} \right)$$

$$F_{55} = \frac{2x_3x_4A}{B^2}$$

$$F_{56} = \frac{-2x_3^2A}{B^2}$$

$$F_{63} = \frac{A}{B^2} \left[-2x_4x_6 + \frac{4x_3x_4A}{B} \right]$$

$$F_{64} = \frac{A}{B^2} \left(3x_4x_5 - x_3x_5 + \frac{4x_4^2A}{B} \right)$$

$$F_{65} = \frac{2x_4^2A}{B^2}$$

$$F_{66} = \frac{-2x_3x_4A}{B^2}$$

$$A = (x_3x_6 - x_4x_5)$$

$$B = (x_3^2 + x_4^2)$$

4.3 Measurement Model

The measurement model employed corresponds to the form of a correlation algorithm developed by Rogers (16:53,63) which provides measurements to a linear Kalman filter measurement model. This algorithm is enhanced over the standard correlation algorithm by the following (19:58,59):

1. The current FLIR data frame is correlated with an estimate of the target's intensity function, an adaptively constructed template, as opposed to the previous frame.
2. Rather than simply outputting the peak of the correlation function, the enhanced correlation algorithm outputs the center of mass of that portion of the correlation function which is greater than some predetermined lower bound, or threshold. This technique prevents the difficulty of having to distinguish global peaks from local peaks, as do the peak-finding algorithms of many conventional correlator algorithms. It is also computationally less burdensome than a peak-finding method.

3. The FLIR/laser pointing commands are generated according to the Kalman filter's propagated state estimate rather than to the output of a standard correlation algorithm, thereby incorporating knowledge of dynamics over the ensuing sample period.

The resultant outputs, or "pseudo-measurements", of the enhanced correlation algorithm are the linear offsets, x_c and y_c , of the centroid with respect to the center of the field-of-view, which are subsequently incorporated into the linear Kalman filter update cycle. A description of this algorithm is presented in the next two sections and a more complete description is discussed in (16:52,70).

4.3.1 Generating the Template. A template of an estimate of the target's intensity profile is created over the previous "N" centered target intensity functions (the shifting property of Fourier transforms is the mechanism which centers the intensity functions on the FLIR plane before averaging). The memory size, "N", over which the intensity functions are averaged is dependent on how the shape functions are varying in time. Highly dynamic intensity functions demand small "N" values while slowly varying intensity functions admit large "N" values.

Since a viable software algorithm concern for any application is the minimization of memory requirements, the finite memory averaging is approximated through the use of an exponential smoothing technique. The exponential smoothing technique resembles the properties of a finite memory filter (10:33,39); however, it requires the storage in

memory of only one FLIR data frame and not "N" frames explicitly. The template for the exponential smoothing technique is maintained by:

$$\hat{I}(t_i) = \gamma I(t_i) + (1 - \gamma) \hat{I}(t_{i-1}) \quad (4-13)$$

where:

$\hat{I}(t_i)$ = template-generated "smoothed" estimate of the target's intensity function

$I(t_i)$ = current FLIR data frame's "raw" intensity function

γ = smoothing constant; $0 < \gamma < 1$

The smoothing constant, γ , in the exponential smoothing algorithm functions as does "N" in the finite memory filter. Large values of γ emphasize the current data frame as would a corresponding small "N" value. A smoothing constant of $\gamma = 0.1$ had previously been determined to suffice for this application (18:V-12).

Figure 4.1 shows a block diagram of the template generation arrangement enclosed within the dashed line. The raw FLIR data array is transformed into the Fourier domain through the use of a Fast Fourier Transform (FFT) implemented using the Cooley-Tukey FFT technique (7:18). The data array is then centered on the FLIR plane by phase shifting it an amount equal to the displacement (in the original untransformed spatial coordinates) of the target and atmospheric position states from the center of the FLIR FOV by the following shifted amount:

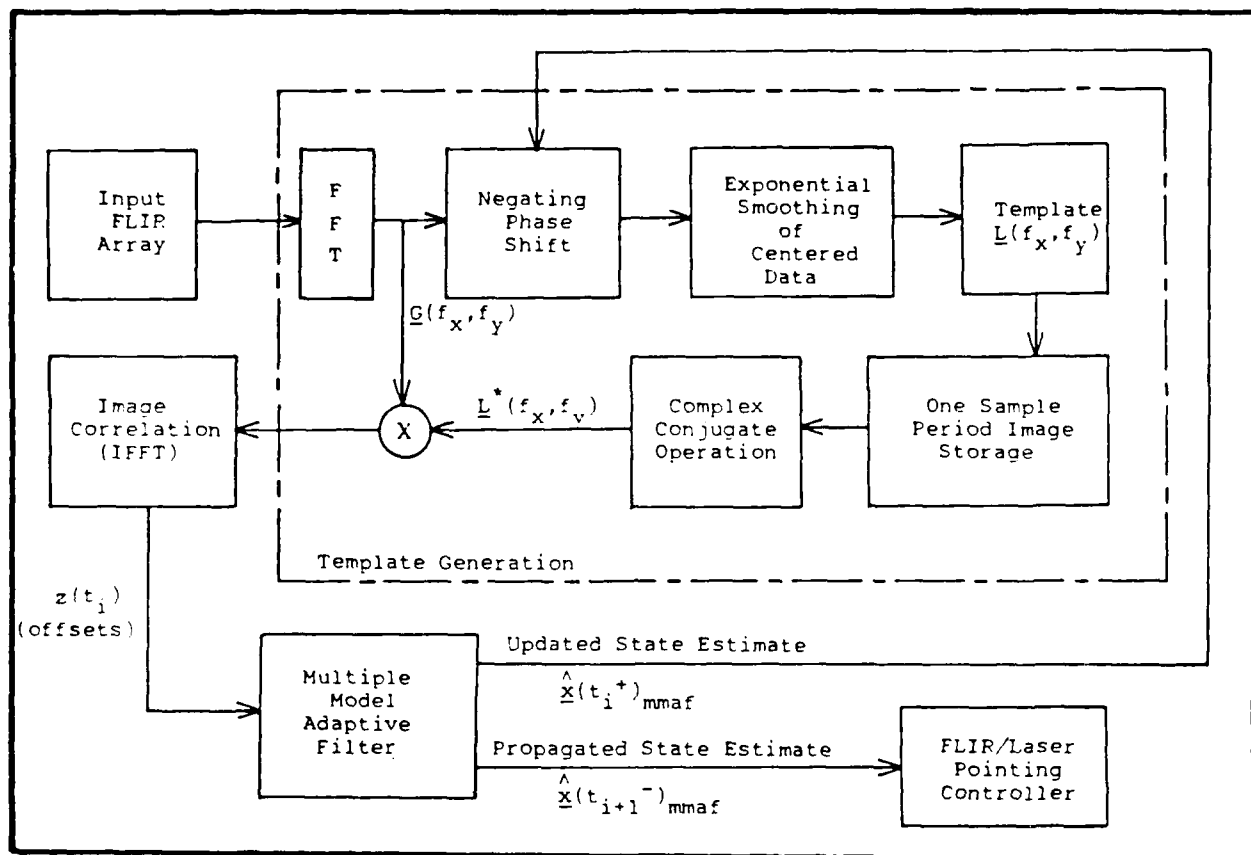


Figure 4.1. Enhanced Correlator/Linear Measurement Model

$$x_{\text{shift}}(t_1) = x_1(t_1^+)_{\text{mmaf}} + x_7(t_1^+)_{\text{mmaf}} \quad (4-14)$$

$$y_{\text{shift}}(t_1) = x_2(t_1^+)_{\text{mmaf}} + x_8(t_1^+)_{\text{mmaf}} \quad (4-15)$$

The shifting property of Fourier transforms, which states that a translational shift in the spatial domain corresponds to a linear phase shift in the frequency domain enables the shift to take place in the frequency domain, or:

$$F\{g(x-x_{\text{shift}}, y-y_{\text{shift}})\} = G(f_x, f_y) \exp\{-j2\pi(f_x \cdot x_{\text{shift}} + f_y \cdot y_{\text{shift}})\} \quad (4-16)$$

where:

$q(x,y)$ = 2-dimensional spatial data array

$F\{\cdot\}$ = Fourier transform operator

$Q(f_x, f_y) = F\{q(x,y)\}$

After centering the data array by the phase shifting technique, the data is subjected to the exponential smoothing algorithm of Equation (4-13). The "smoothed" data is then stored in the form of a template and correlated with the subsequent FLIR data frame which creates the "pseudo-measurements".

4.3.2 "Pseudo-Measurements" by Enhanced Correlation.

The enhanced correlation algorithm creates the "pseudo-measurements" in the form of position offsets in two orthogonal directions. These offsets represent the distance between the center of the FLIR FOV and the centroid of the IR image. The current FLIR data frame is correlated in the Fourier domain with the template generated from the previous sample time, thus resulting in a cross correlation. This cross correlation is performed by computing the Inverse Fast Fourier Transform (IFFT) of the following:

$$F\{q(x,y)*l(x,y)\} = Q(f_x, f_y)L^*(f_x, f_y) \quad (4-17)$$

where:

$q(x,y)*l(x,y)$ = cross correlation of $q(x,y)$ and $l(x,y)$

$q(x,y)$ = measured target intensity function,
current FLIR data frame

$l(x,y)$ = expected target intensity function,
generated in the form of a template

$$Q(f_x, f_y) = F\{q(x, y)\}$$

$$L^*(f_x, f_y) = \text{complex conjugate of } F\{l(x, y)\}$$

After performing the IFFT, a thresholding is performed on the correlation function, $q(x, y) * l(x, y)$, that effectively zeros out any value that is less than 30% of the function's maximum value (14:52,55). The resultant function corresponds to the relative displacement, or offset, of the center-of-intensity between the current FLIR data frame and the template. These offsets are due to the effects of target dynamics, atmospheric jitter, and measurement noise (bending effects are ignored in the filters, however, the variances of the measurement noises can be increased during filter tuning to account for this), or:

$$x_{\text{offset}} = x_t + x_a + n_x \quad (4-18)$$

$$y_{\text{offset}} = y_t + y_a + n_y \quad (4-19)$$

Equations (4-18) and (4-19) can be represented in state space form by:

$$z(t_1) = H_f x_f(t_1) + v_f(t_1) \quad (4-20)$$

where:

$$z(t_1) = (x_{\text{offset}} \ y_{\text{offset}})^T, \text{ measured in pixels}$$

$$x_f(t_1) = \text{filter state vector of Equation (4-1)}$$

$$H_f = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4-21)$$

$\underline{v}_f(t_i)$ = 2-dimensional, discrete-time, zero-mean, white Gaussian measurement corruption noise measured in pixels, of covariance \underline{R}

The measurement noise, $\underline{v}_f(t_i)$, encompasses the spatially correlated background noise, the FLIR sensor error, errors due to ignoring the vibration/bending effects, and errors due to the FFT/IPFT processes. The covariance matrix associated with this error (except for the impact of the ignored bending effects) has been determined empirically to be (16:63,68):

$$\underline{R} = \begin{bmatrix} 0.00436 & 0 \\ 0 & 0.00598 \end{bmatrix} \quad (4-22)$$

4.4 Summary

This chapter has formulated the filter and measurement models for the MMAF's elemental filters. One filter model represents the target's acceleration as a first order Gauss-Markov process, while the other filter model treats the target's trajectory as a series of constant turn-rate segments. Both dynamics models utilize a common linear measurement model that generates "psuedo-measurements" via a cross-correlation technique between a template generated from the previous data frame and the current data frame. An advantage to the linear measurement model is the ability to employ the linear Kalman filter update equations (9:117,118) rather than algorithms which are more computationally burdensome.

V. Tracking Algorithm

5.1 Introduction

This chapter embodies the concepts presented in the preceding three chapters and describes the tracking algorithm employed in this research endeavor. Following the discussion on the overall tracking algorithm, the mechanisms exercised in evaluating the algorithm are also discussed.

5.2 Overview of the Tracking Algorithm

A Bayesian Multiple Model Adaptive Filter including a bank of five independently operating Kalman filters provides the basis for the tracking algorithm. The processed fields-of-view for each filter, along with the assumed target's dynamics in the FLIR plane directions, appear in Table 5.1. Note that filters #1, #2, and #3 are represented by square fields-of-view, whereas filters #4 and #5 are rectangular field-of-view filters.

Table 5.1

MMAF Elemental Filters

<u>Filter #</u>	<u>FOV (pixels)</u>	<u>Assumed x-dynamics</u>	<u>Assumed y-dynamics</u>
1	8 X 8	benign	benign
2	24 X 24	20 g	20 g
3	8 X 8	10 g	10 g
4	24 X 8	20 g	benign
5	8 X 24	benign	20 g

5.2.1 Gauss-Markov Model Tracker. The five elemental filters in the Gauss-Markov MMAP (GM MMAP) are based upon the Gauss-Markov acceleration model of Section 4.2.1. Each filter was tuned with the appropriate direction-dependent acceleration variance, σ^2 , and time constant, τ , (14:68). These parameters are shown in Table 5.2 for each of the five filters. Notice that the parameters pertaining to x- and y-directions for filters #1, #2, and #3 are the same, whereas they differ for filters #4 and #5 due to the rectangular geometry of the FOV of these filters. Filters #4 and #5 are constructed with a larger field-of-view size in the direction in which harsh maneuvers are expected. The filter driving

Table 5.2

Gauss-Markov Filter Parameter Values

<u>Filter #</u>	<u>τ_x</u>	<u>σ_x^2</u>	<u>τ_y</u>	<u>σ_y^2</u>	<u>τ_a</u>	<u>σ_a^2</u>	<u>Δt</u>
1	4.0	250	4.0	250	.0707	.2	1/30
2	1.5	10000	1.5	10000	"	"	"
3	1.5	2000	1.5	2000	"	"	"
4	1.5	10000	4.0	250	"	"	"
5	4.0	250	1.5	10000	"	"	"

Units: $\tau_x, \tau_y, \tau_a, \Delta t$: seconds
 σ_x^2, σ_y^2 : pixels²/seconds⁴
 σ_a^2 : pixels²

noise strength, "Q", is not shown but can easily be computed from the following relationship:

$$Q_{11} = 2\sigma^2/\tau \quad (5-1)$$

where:

σ^2 = the filter's assumed variance for that target acceleration process or jitter position process

τ = correlation time of target acceleration or of jitter

The parameters shown in Table 5.2 for filters #2 and #3 were determined by tuning a single, large FOV and small FOV Kalman filter, respectively. Using a trajectory #2 target maneuver (reference Section 3.4.1), filter #2 was tuned to 20-g dynamics and filter #3 was tuned to 10-g dynamics. Parameter values for filters #4 and #5 are chosen such that the highly-varying dynamics coincide with the appropriate channel direction of the rectangular FOV. The jinking maneuvers correspond to the elongated direction of the FOV and the short end of the FOV is characterized by the benign dynamics of filter #1. Also note that the sample period, Δt , for past research efforts has been 30 Hz. Section 6.3 addresses the potential for implementing the tracking algorithm at a 50 Hz rate.

5.2.2 Constant Turn-Rate Model Tracker. The MNAF structure of the constant turn-rate (CTR) model tracker is of the same configuration as shown in Table 5.1. The key difference between the CTR and GM model trackers is the strength of the driving white noise terms, "Q", as shown in

Table 5.3 for the CTR tracker (the computational burden of the CTR model is unnecessary for the benign dynamics of filter #1). Notice that the "Q" values do not correspond to the relation described in Equation (5-1). Previous research has shown that a CTR filter is considered to be adequately tuned to a trajectory #2 maneuver when the peak mean-error is approximately 1.5 times the filter computed error standard deviation or root-mean-squared (RMS) value at a given time, rather than when the filter's computed RMS errors match the actual RMS errors committed throughout the simulation, as evaluated over a 10-run Monte Carlo simulation (10:69,71). The consequence of this relationship is that the CTR filter overestimates its own errors more than does the GM filter.

Table 5.3
CTR Filter Parameter Values

<u>Filter #</u>	<u>Q_x</u>	<u>Q_y</u>
2	15000	15000
3	4000	4000
4	15000	125
5	125	15000
Units: pixels ² /seconds ⁵		

5.3 Field-of-View Processing

The fields-of-view for all the filters are represented as 8 X 8 data arrays in order to exploit the single generic correlation algorithm. This is accomplished by redefining the pixel size in filters #2, #4, and #5 for correlation processing (filters #1 and #3 already are parameterized in an 8 X 8 array). Filter #2's FOV is represented as an 8 X 8 data array by partitioning the 24 X 24 array into an 8 X 8 array of 3 X 3 elements, where each 3 X 3 element consists of an average of the nine "small" (20 micro-radians on a side) elements from the larger FOV. This essentially simulates a data array of a "large" (60 micro-radian) pixel array. The redefined pixel for filter #4 is averaged over a 3 X 1 array of "small" pixels due to the rectangular geometry and measures 60 micro-radians x 20 micro-radians. By similar argument, filter #5's redefined pixel measures 20 micro-radians x 60 micro-radians. The FOVs and unit pixel sizes (shaded regions) for correlation processing are shown in Figures 5.1 and 5.2.

5.4 Filter Parameters

At initialization of the tracking simulation, accurate a priori knowledge of velocity is assumed and position is acquired by an algorithm developed by Tobin (19:130,139). The position states, x_1 and x_2 , are initialized by positioning the target's center of mass in the center of the FLIR FOV. The velocity states, x_3 and x_4 , are initialized

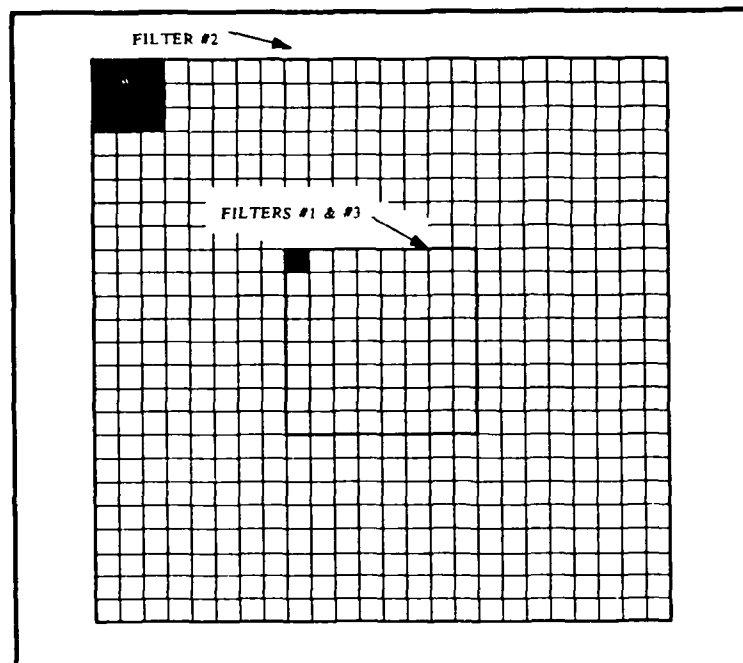


Figure 5.1. Square Fields-of-View, Filters #1, #2, and #3

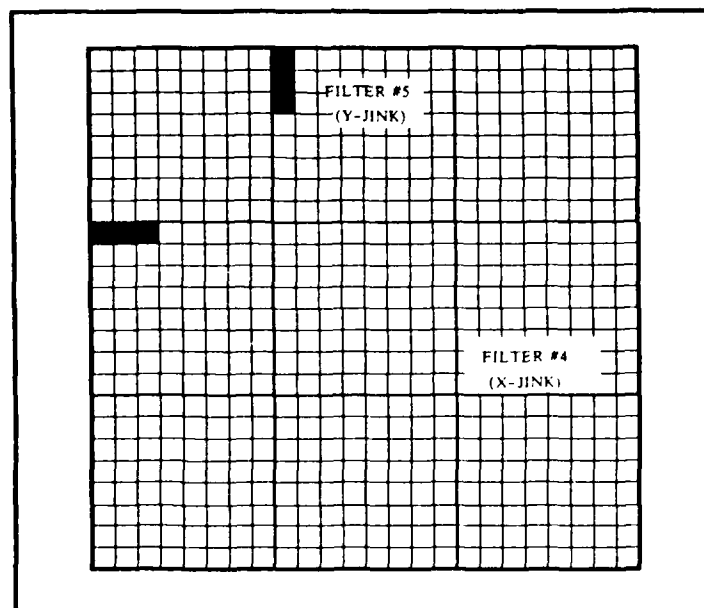


Figure 5.2. Rectangular Fields-of-View, Filters #4 and #5

dependent upon the trajectory selection and the acceleration states, x_5 and x_6 , are initialized by computing the delta velocities at time t_0 and time t_1 and dividing by the sample period. The atmospheric jitter states, x_7 and x_8 , are initially zero. The error covariance matrix, $P(t_0)$, is initially reflected as:

$$P(t_0) = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 \end{bmatrix} \quad (5-2)$$

The conditional probabilities, P_1 , P_2 , P_3 , P_4 , and P_5 , are initialized to 0.96, 0.01, 0.01, 0.01, and 0.01 for filters #1 thru #5, respectively. The initial "heavy" weighting of the conditional probability on filter #1 is due to the "best-matching" characteristics of filter #1's benign dynamics to the target's straight and level trajectory for time less than two seconds. In actual practice, initial acquisition can be presumed to occur on a benign target.

In the event of loss of track due to gross errors in any of the filter's target position estimates, the tracking algorithm executes a reacquisition routine. This routine

resets the state vector and covariance matrix of the divergent filter as a linear combination of the state vectors and covariance matrices of the nondivergent filters. The conditional probabilities of the nondivergent filters are then scaled to sum up to one prior to evaluating the state vector and error covariance matrix of the divergent filters by :

$$\hat{\underline{x}}_d = \sum_{n=1}^N p_n \hat{\underline{x}}_n \quad (5-3)$$

$$\underline{P}_d = \sum_{n=1}^N p_n (\underline{P}_n + (\hat{\underline{x}}_n - \hat{\underline{x}}_d)(\hat{\underline{x}}_n - \hat{\underline{x}}_d)^T) \quad (5-4)$$

where:

$\hat{\underline{x}}_d, \underline{P}_d$ = state vector and error covariance matrix of each divergent filter

$\hat{\underline{x}}_n, \underline{P}_n$ = state vector and error covariance matrix of each nondivergent filter

N = number of nondivergent filters

p_n = hypothesis conditional probability of the n -th nondivergent filter. The p_n values are scaled such that: $p_1 + p_2 + \dots + p_N = 1.0$

5.5 Tracking Algorithm Statistics

A Monte Carlo simulation provides the mechanism for evaluating the tracking algorithms' performance. Based upon the conclusions of previous research efforts that the sample statistics of ten Monte Carlo simulation runs provides sufficient convergence to the actual statistics from an infinite number of runs (19:80), the statistics from ten

Monte Carlo runs will act as the basis upon which tracker performance is evaluated.

The sample mean errors of the tracking algorithm's estimates are computed by :

$$\begin{aligned}\overline{E_{xd}}(t_i) &= 1/N \sum_{n=1}^N e_{xdn}(t_i) \\ &= 1/N \sum_{n=1}^N [x_{dn}(t_i) - \hat{x}_{dnf}(t_i)]\end{aligned}\quad (5-5)$$

where:

$\overline{E_{xd}}(t_i)$ = sample mean error of the x target position estimate at time t_i , averaged over N runs

$e_{xdn}(t_i)$ = error in the MMAP x-position estimate at t_i during simulation n

$\hat{x}_{dnf}(t_i)$ = MMAP estimate of target's x-position at t_i during simulation n

$x_{dn}(t_i)$ = truth model value of the x-position of the target at t_i during simulation n

N = number of Monte Carlo runs

The sample variance of the error, defined in terms of the same parameters as the sample mean errors, is described by:

$$\sigma_{xd}^2(t_i) = [1/(N-1)] \sum_{n=1}^N e_{xdn}^2(t_i) - (N/(N-1)) \overline{E_{xd}}^2(t_i) \quad (5-6)$$

Two error parameters are of particular concern when evaluating the tracking algorithm. The first is the error committed in estimating the target's position in both the x- and y- FLIR plane directions. This parameter serves as the fundamental evaluation mechanism for tracker performance. The second parameter is the estimation error committed in

locating the target's centroid of the FLIR plane. Because the target's centroid location is paramount for centering it on the FLIR plane for the template generation, this error parameter supplies information with respect to the filters' ability to reconstruct the target's shape function and centroid location.

5.6 Performance Plots

Eight performance plots are utilized in this research effort to evaluate filter performance. The performance plots for both the x- and y-direction in the FLIR plane are as follows :

1. True x-position RMS error vs. filter-computed x-position RMS error
2. True y-position RMS error vs. filter-computed y-position RMS error
3. Mean x-target position error, \pm one standard deviation, plotted at t_1^- for all i
4. Mean y-target position error, \pm one standard deviation, plotted at t_1^- for all i
5. Mean x-target position error, \pm one standard deviation, plotted at t_1^+ for all i
6. Mean y-target position error, \pm one standard deviation, plotted at t_1^+ for all i
7. Mean x-centroid error, \pm one standard deviation, plotted at t_1^+ for all i
8. Mean y-centroid error, \pm one standard deviation, plotted at t_1^+ for all i

Performance plots #1 and #2 provide information with respect to the tuning process, plots #3 thru #6 provide

tracking performance evaluation, and plots #7 and #8 provide information regarding the effectiveness of the target intensity template identification.

Illustrations of plots #2, #4, #6, and #8 are shown in Figures 5.3 thru 5.6, respectively, for the Gauss-Markov MMAP algorithm tracking a 10-g, trajectory #2 target maneuver (note that the expected level of bending is included in the truth model but not in the filter model). Each of these plots portrays the evidence of the target maneuver at $t = 2.0$ seconds.

Each performance plot is labelled with a plot designation code that identifies the specific simulation scenario for which the data is referenced. This code is in the following format:

field 1/field 2/field 3/field 4/field 5/field 6

where fields #5 and #6 are optional, and the fields contain the following information:

field 1 :	GM	: Gauss-Markov filter model
	CTR	: constant turn-rate filter model
field 2 :	MMAP	: multiple model adaptive filter
	SNG-8	: single filter with small (8x8) FOV, used to establish the benchmark
field 3 :	T(#)	: identifies type trajectory (#1, #2, #3, #4, #5, or #6)
field 4 :	(#)-G	: identifies the magnitude (in g's) of the target maneuver

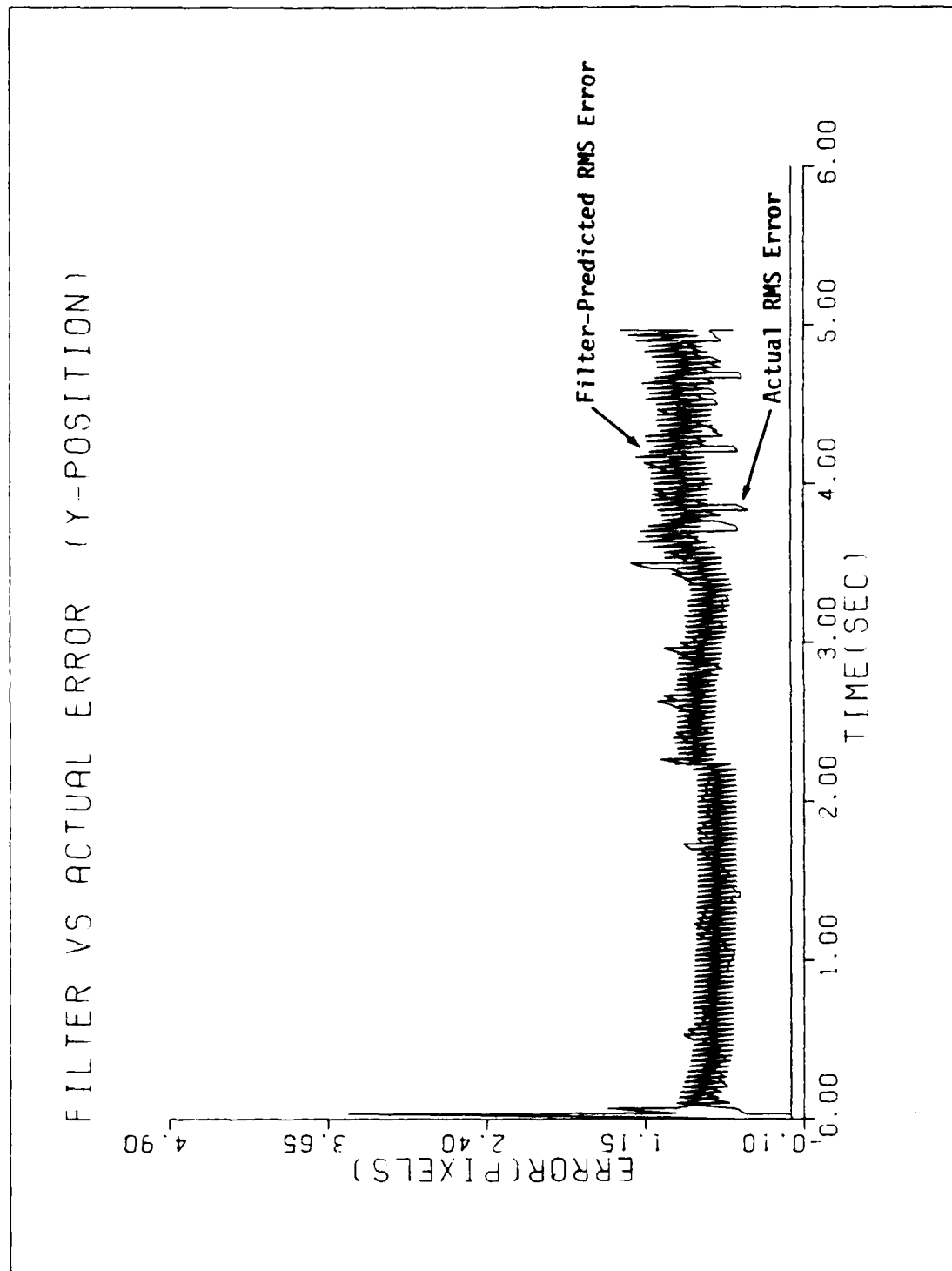


Figure 5.3. Performance Plot; GM/MMAF/T2/10-G/QB1;
Actual RMS Error vs Filter-Computed RMS Error

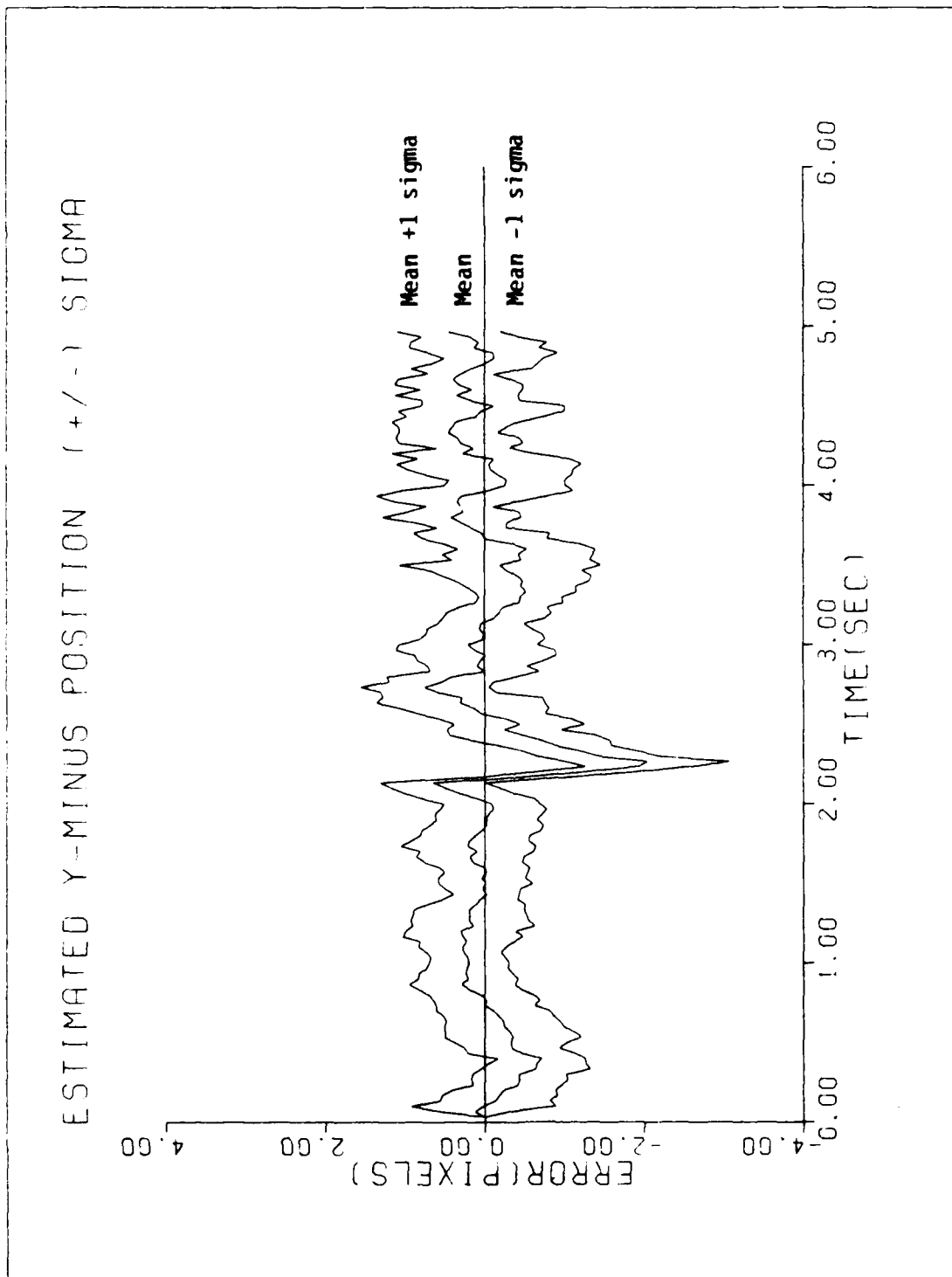


Figure 5.4. Performance Plot; GM/MMAF/T2/10-G/QB1;
Mean Position Error $\pm \sigma$, at t_1 -

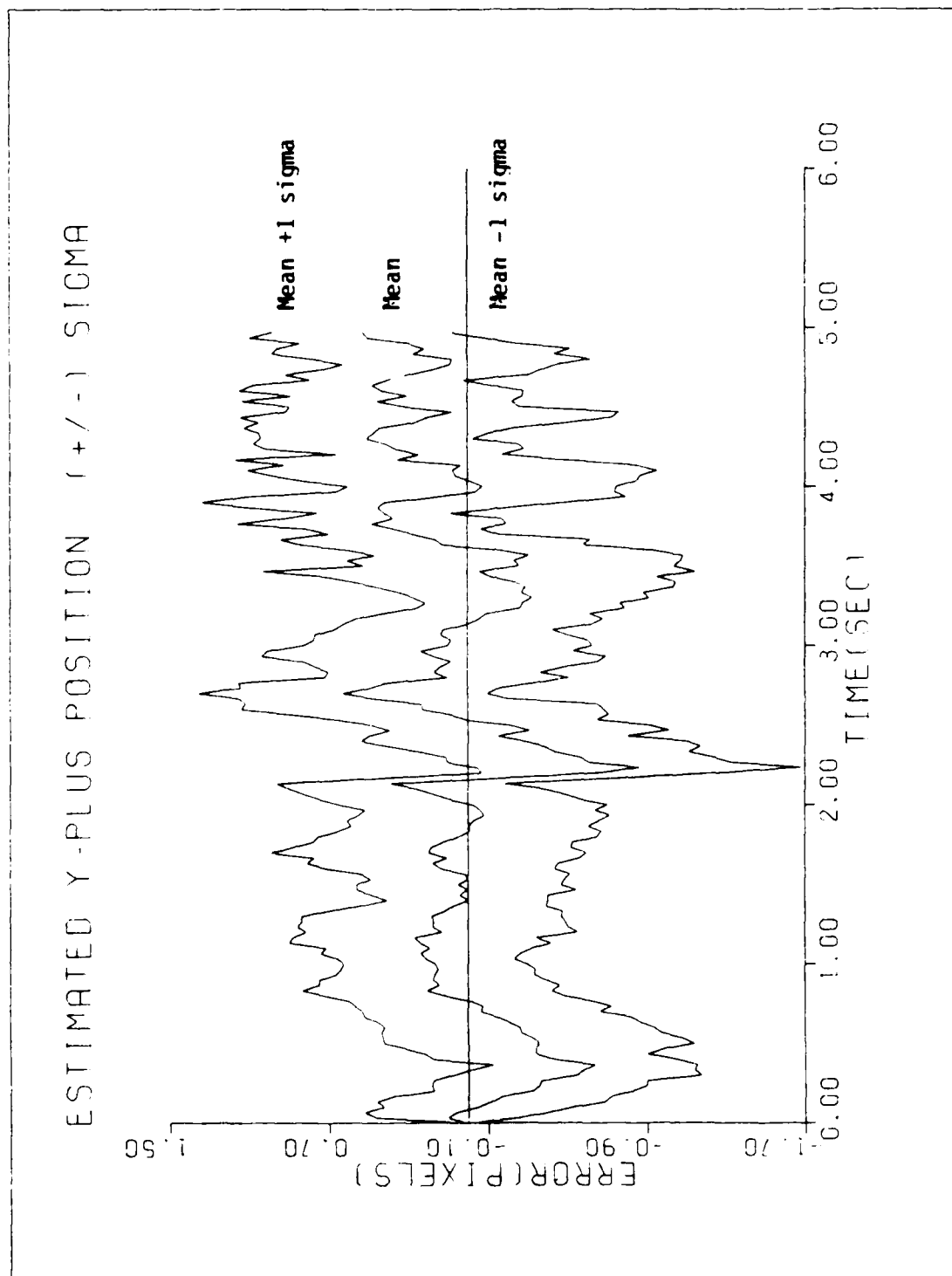


Figure 5.5. Performance Plot; GM/MMAF/T2/10-G/QB1;
Mean Position Error $\pm \sigma$, at $t_1 +$

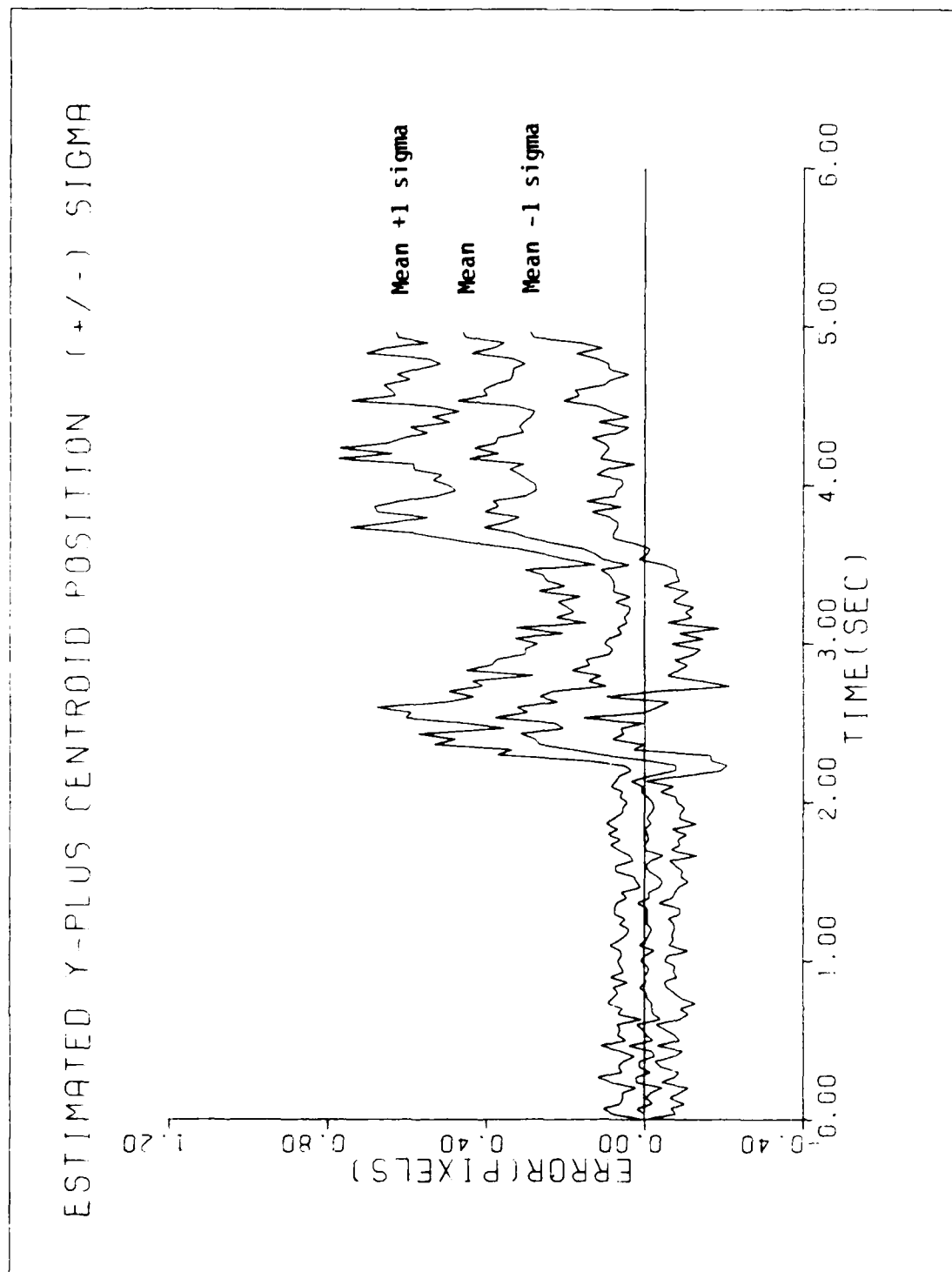


Figure 5.6. Performance Plot; GM/MMAF/T2/10-G/QB1;
Mean Centroid Error $\pm \sigma$, at $t_1 +$

field 5 : (Optional)

$\tau = \tau_1, \tau_2$: identifies the assumed filter correlation time pairs when unlike the nominal (4,1.5)

50 Hz : identifies a simulation with the 50 Hz sample time incorporated

QB(#) : identifies the level of bending phenomena included in the truth model

$z_0 =$: identifies when the initial z-coordinate is different from the nominal (20,000 meters)

field 6 : KP = : identifies the different pixel proportionality constant size from nominal (20 μ rad)

R(+) : identifies when the measurement variance, R, is different from that identified in Equation (4-22)

For example, the plot designation codes appearing in Figures 5.3 thru 5.6 reflect:

GM/MMAF/T2/10-G/QB1

This designation implies that the simulation involved the Gauss-Markov MMAF tracking model with the expected level of bending included, against a trajectory #2 target maneuver with the target pulling a 10-g turn.

5.7 Greyscales

Greyscales are symbols used to characterize the ranges on FLIR plane images and the filter parameters. The symbols are shown in Table 5.4. The greyscale

AD-A190 619

A MULTIPLE MODEL ADAPTIVE TRACKING ALGORITHM AGAINST
AIRBORNE TARGETS(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING

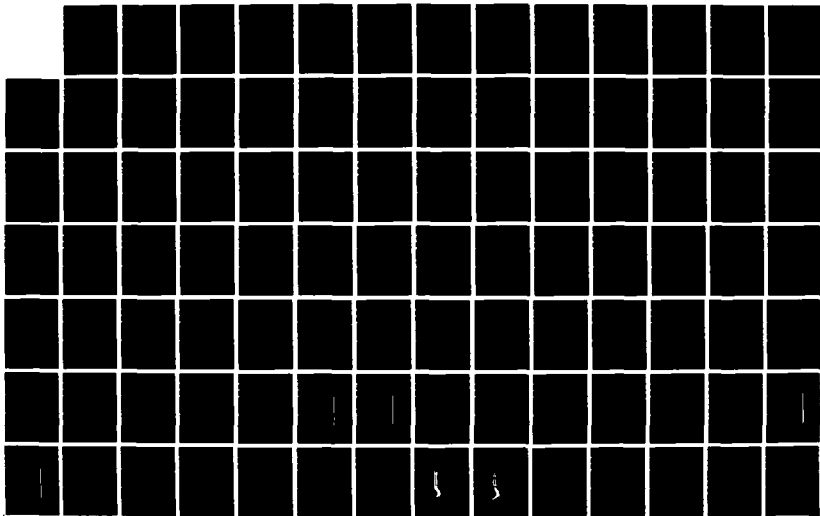
2/4

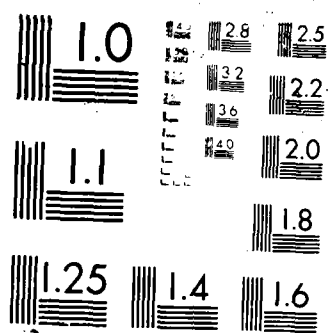
UNCLASSIFIED

T A LEENEY DEC 87 AFIT/GE/ENG/87D-37

F/G 17/5.1

ML





which is a pictorial view of greyscale symbols, illustrates the adaptive identification of the target's intensity shape function in the form of a template on a 24-by-24 grid of pixels. The example shown in Figure 5.7 is the result of a comparison between a template and a noise-corrupted FLIR measurement array. The spread, or arrangement of the greyscale symbols indicate the dispersion of the hotspot about its centroid and also the location of this intensity peak relative to the center of the FLIR FOV.

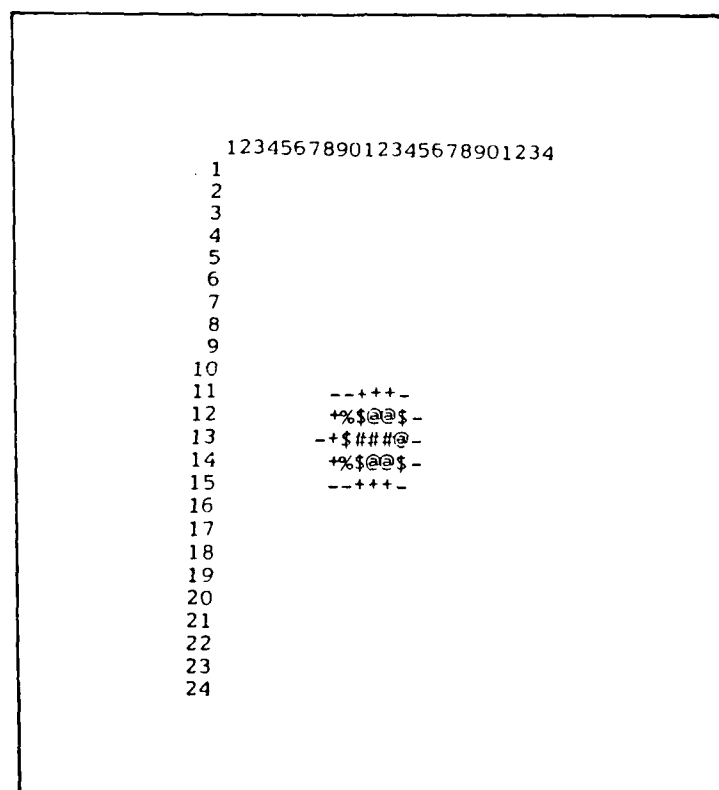


Figure 5.7. Noise Corrupted FLIR Measurement Array

Table 5.4
Greyscale Symbol Key

<u>Symbol</u>	<u>Intensity Units</u>
(blank)	$0 \leq I \leq 14$
-	$14 \leq I \leq 28$
+	$28 \leq I \leq 42$
%	$42 \leq I \leq 56$
\$	$56 \leq I \leq 70$
@	$70 \leq I \leq 84$
#	$84 \leq I$

5.8 Summary

This chapter has blended the ideas and concepts discussed in the previous three chapters into a viable multi-filter tracking algorithm. As a basis of comparison, two distinct tracker models have been proposed - a Gauss-Markov acceleration model tracker and a constant turn-rate model tracker, both of which utilize the same FOV processing algorithms. These algorithms, although normally implemented optically, represent the FLIR-generated information in the form of digitally generated intensities that can be portrayed via greyscale diagrams, which serve as the templates as discussed in Section 4.3.1. Finally, the tracker statistics, through which the evaluations in Chapter VI will be analyzed, were presented.

VI. Performance Analysis

6.1 Introduction

This chapter presents the results of the various analyses for those topics outlined in Section 1.3. The evaluation of the benefits of simulating the tracking algorithms at a 50 Hz rate versus a 30 Hz rate are discussed in Section 6.2. After the effects of the bending/vibration states are analyzed utilizing a single filter for both the Gauss-Markov model tracker and the constant turn-rate model tracker in Section 6.3, the next section then analyzes the effects of bending/vibration effects with the Gauss-Markov multiple model adaptive filter algorithm. A sensitivity study of various parameters for which the filters are not retuned (other than just altering that specific parameter in the filter) is presented in Section 6.5, and a sensitivity analysis of two additional target trajectories is presented in Section 6.6.

6.2 50 Hz Implementation Analysis

As suggested in Section 1.3.2, the performance gains attributable to a 50 Hz implementation are investigated. This assessment is performed by first establishing a benchmark of performance for both the Gauss-Markov and constant turn-rate tracker models simulated at the presently-configured sample rate of 30 Hz. After the benchmark has been established, the appropriate software changes are implemented to accommodate the faster sampling frequency,

and an identical simulation scenario as conducted with the benchmark runs is performed, indicating the improvement or degradation as a result of sampling at a higher frequency.

As mentioned in Section 1.3, Tobin's results (13) form the basis of the tracking performance benchmark for the analyses to be presented in this chapter. This benchmark is defined as the performance of a single filter which is artificially "told" about certain dynamic parameters at the onset of a maneuver. The filter employed for the benchmark simulations is a small FOV (8 x 8) filter for a trajectory 2, 10-g maneuver. The changing parameter for this case is the strength of the filter driving noise, "Q", which is increased prior to the onset of a maneuver so that the filter-computed RMS error reaches a steady-state value by the time the maneuver occurs, thus allowing the filter bandwidth to open by the time the maneuver is initiated. These simulations are conducted for both the Gauss-Markov tracker and the constant turn-rate tracker models.

6.2.1 Gauss-Markov Model. The performance plots for the benchmark run and for the case of the 50 Hz simulation are shown in Figures C-1 through C-8, and C-9 through C-16, respectively. Notice that each set of performance plots, as is true of all subsequent simulations, is composed of eight different types of plots. The first two, as in Figures C-1 and C-2, show the filter-computed estimation error standard deviation compared to the actual RMS error (in pixels) for

both the x- and y-directions, respectively. These two plots indicate the tuning adequacy of the filter(s) by showing a comparison between the actual and the filter-computed RMS values. Plots three and four, as in Figures C-3 and C-4, identify the actual estimation mean error \pm one standard deviation at time = t_1^- for the x- and y-directions, respectively. These parameters are used as a basis for control generation. Plots five and six, as in Figures C-5 and C-6, identify the same information as plots three and four, but at time = t_1^+ . These parameters provide the best possible filter(s) estimation accuracy. The initiation of the target maneuver at two seconds into the simulation is evident in Figures C-4 and C-6. Plots seven and eight, as shown in Figures C-7 and C-8, show the actual centroid position estimation error mean \pm one standard deviation at time = t_1^+ for the x- and y-directions, respectively. These parameters provide information regarding the adequacy of image centering to aid in the construction of the target image template.

In both the 30 Hz and the 50 Hz case, the dynamics driving noise strength, "Q", is increased at frame #55 for the 30 Hz simulation and at frame #195 for the 50 Hz simulation, allowing for a five-frame transient prior to target maneuver initiation. Tables 6.1 and 6.2 show the temporally averaged filter error statistics (mean and standard deviation, or "sigma"), the filter's peak-mean y-position estimate errors and the recovery time for both

cases (x-position is just as important; however, the maneuvers evaluated here exhibit harshly changing y-direction characteristics). The two time-averaged intervals, 0.5 to 2.0 and 3.5 to 5.0 seconds, are chosen to allow a 0.5 second minimum for transients to die out. These intervals are selected to show comparable statistical data during the benign region and during the maneuver. The peak-mean error is defined as the greatest mean-error occurring during a given simulation and the recovery time is defined as the time it takes for the filter to recover, or reach steady state, from the maneuver. Both of these parameters are precise only to the degree that the values are discernible from the plots (Figures C-4, C-6, C-12, and C-14), and are considered accurate to within five percent. In these and subsequent tables, the heading designations correspond to the convention established in Section 5.6.

In comparing the temporally averaged mean and one sigma values for the two tracking simulations, Tables 6.1 and 6.2, reveals a slight improvement in the mean error associated with estimating the position and an improvement in the true standard deviation about that mean error, which is expected since the algorithm samples the data at a faster rate and thus allows less time for error growth. This improvement in error standard deviation about the mean can be seen in Figures C-3 through C-6 and C-11 through C-14 by comparing the relative displacement of the one sigma curves relative

Table 6.1

Single Filter Benchmark;
GM/SNG-S/T2/10-G

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$	-.1623 / .4089	.2073 / .4912
$\hat{y}_{err}(t_1^-)$.0136 / .4241	-.2588 / .4941
$\hat{x}_{err}(t_1^+)$	-.1044 / .3765	.1479 / .4290
$\hat{y}_{err}(t_1^+)$.0031 / .3876	-.0998 / .4240
$\hat{x}_{-cent_{err}}(t_1^+)$.0068 / .1226	.0007 / .1546
$\hat{y}_{-cent_{err}}(t_1^+)$	-.0083 / .0706	.0577 / .0714
y peak-mean error (t_1^-) = -2.0 pixels		
y peak-mean error (t_1^+) = -1.4 pixels		
recovery time = .7 seconds		

Table 6.2

Single Filter Simulation;
GM/SNG-S/T2/10-G//50 Hz

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$	-.0385 / .3779	.1110 / .4557
$\hat{y}_{err}(t_1^-)$.0125 / .3787	-.2352 / .4397
$\hat{x}_{err}(t_1^+)$	-.0041 / .3568	.0776 / .4172
$\hat{y}_{err}(t_1^+)$.0096 / .3522	-.1486 / .3973
$\hat{x}_{-cent_{err}}(t_1^+)$.0400 / .0889	.0504 / .0911
$\hat{y}_{-cent_{err}}(t_1^+)$.0169 / .0795	.0719 / .0815
y peak-mean error (t_1^-) = -2.0 pixels		
y peak-mean error (t_1^+) = -1.3 pixels		
recovery time = .6 seconds		

to the mean curve in the two sets of plots (also note the difference in scales between the two sets of plots).

6.2.2 Constant Turn-Rate Model. The performance plots for the two simulations, the benchmark run at 30 Hz and the 50 Hz run, are shown in Figures C-17 through C-24 and C-25 through C-32, respectively. Notice that the filter tuning for the CTR model is much more conservative than is the case with the GM model. Since the filters with the CTR dynamics model exhibit a longer filter-computed covariance matrix transient than do the GM filters, "Q" was increased at frame #35 (frame #175 in the 50 Hz case) to allow the transient to reach steady state and the filter bandwidth to open appropriately prior to the maneuver initiation. Tables 6.3 and 6.4 show the statistical data for the two simulations, which show results that are comparable to those of the Gauss-Markov tracker. In comparing the filter estimated error curves for both the simulations, Figures C-25 and C-26 reveal that the relative displacement of the filters' estimated errors between time t_1^- and t_1^+ is larger than what is observed in the 30 Hz case of Figures C-17 and C-18. This occurs due to changing the continuous-time dynamics noise strength, "Q", for the simulations to allow the discrete-time noise covariance, "Q_d", to remain the same. The "Q" relationship is described by the following approximation inherent to the CTR model:

$$Q_d \approx Q\Delta t \quad (6-1)$$

Table 6.3

Single Filter Benchmark;
CTR/SNG-S/T2/10-G

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.4106 / .3905	.2343 / .3881
$\hat{y}_{err}(t_i^-)$	-.0223 / .3822	.3372 / .4232
$\hat{x}_{err}(t_i^+)$.3406 / .3715	.1872 / .3588
$\hat{y}_{err}(t_i^+)$	-.0242 / .3570	.2987 / .3829
$\hat{x}_{cent_{err}}(t_i^+)$.0009 / .1145	-.0203 / .1163
$\hat{y}_{cent_{err}}(t_i^+)$	-.0156 / .0708	.1347 / .0764
y peak-mean error (t_i^-) = -6.0 pixels		
y peak-mean error (t_i^+) = -4.0 pixels		
recovery time = 1.3 seconds		

Table 6.4

Single Filter Simulation;
CTR/SNG-S/T2/10-G//50 Hz

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.5956 / .3160	.1049 / .3985
$\hat{y}_{err}(t_i^-)$.0164 / .3069	.3702 / .3618
$\hat{x}_{err}(t_i^+)$.5461 / .3058	.0744 / .3805
$\hat{y}_{err}(t_i^+)$.0149 / .2943	.3389 / .3435
$\hat{x}_{cent_{err}}(t_i^+)$.0084 / .0826	-.0402 / .0919
$\hat{y}_{cent_{err}}(t_i^+)$.0232 / .0771	.1962 / .0799
y peak-mean error (t_i^-) = -6.0 pixels		
y peak-mean error (t_i^+) = -5.0 pixels		
recovery time = 1.2 seconds		

This relationship effectively decreases the discrete-time noise strength as the sample frequency is increased, provided the continuous-time noise strength is unchanged. Although conceptually it makes sense to keep "Q" constant rather than " Q_d ", "Q" was increased to maintain the equivalent discrete-time filter driving noise for this simulation (to maintain a direct comparison between the two algorithms at the different sampling frequency), thus realizing a higher slope between t_1^- and t_1^+ for the filter-computed estimation errors. Since processing at a 50 Hz greatly increases (73% increase) computer processing time, the benefit of decreasing the uncertainty of the position estimate on the order of a tenth of a pixel, as compared to the 30 Hz version, is not warranted. Therefore, in an effort to conserve computer processing time, subsequent simulations are processed with a sample period of 30 Hz.

6.3 Single Filter Bending/Vibration Analysis

This section addresses the impact of including the bending/vibration states of a large space structure into the tracking algorithm's truth model without modifying the tracker's filter model. Thus, this is a robustness study of the original filter design. The two benchmark scenarios, processed at 30 Hz, discussed in Section 6.2, form the baseline of comparison for both the Gauss-Markov tracking model and the constant turn-rate tracking model.

6.3.1. Gauss-Markov Model Tracker. The performance plots for the case of including the bending/vibration states are shown in Figures D-1 through D-8. Notice from Figures C-1, C-2, D-1, and D-2, that the filter estimates for the simulation with the bending/vibration states included and without additional tuning underestimate the actual error. Although the true RMS value of x- and y-position has increased with the inclusion of vibration and bending, the filter-computed error does not appear to change. This implies that the filter is more confident of its outputs than it should be. Although the filter can be improved by retuning, the robustness of the original filter is the prime concern under consideration. In comparing the time-averaged statistics of Tables 6.1 and 6.5, it is evident that, although the mean error between the two cases does not seem to vary much, the one sigma values about the mean have increased by approximately 2/10 to 3/10 pixels with the inclusion of the bending/vibration. This relationship can also be seen in Figures C-3 through C-6 and Figures D-3 through D-6. This increase in the uncertainty with respect to the position variables is due partly to the filter not estimating the bending states and partly to the enhanced correlation algorithm not considering the effects of bending when establishing the offsets from the center of the field-of-view in constructing the image. Recall the measurement equations for the two directions:

$$z_1 = x_d + x_a + x_b + v_1 \quad (6-2)$$

$$z_2 = y_d + y_a + y_b + v_2 \quad (6-3)$$

where these measurements represent the location of the target centroid as indicated by the correlator. However, the filter is not provided any information with respect to the two bending states, x_b and y_b , nor is it tuned with additional measurement uncertainty noise associated with v_1 and v_2 . No appreciable change in estimating the location of the centroid appears between the two simulations.

Table 6.5

Single Filter With Bending/Vibration;
GM/SNG-S/T2/10-G/QB1

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$	-.1449 / .6297	.1536 / .7625
$\hat{y}_{err}(t_i^-)$.0610 / .6407	-.1588 / .6698
$\hat{x}_{err}(t_i^+)$	-.0849 / .5947	.0922 / .7133
$\hat{y}_{err}(t_i^+)$.0474 / .6026	-.0015 / .6096
$\hat{x}_{-cent_{err}}(t_i^+)$	-.0057 / .1195	-.0333 / .1296
$\hat{y}_{-cent_{err}}(t_i^+)$	-.0055 / .0758	.0617 / .0774
y peak-mean error (t_i^-) = -2.0 pixels		
y peak-mean error (t_i^+) = -1.2 pixels		
recovery time = .7 seconds		

6.3.2 Constant Turn-Rate Model Tracker. The performance plots for the case where the bending/vibration states are included using the constant turn-rate model tracker are shown in Figures D-9 through D-16, and the statistical data appears in Table 6.6. As was evident in the Gauss-Markov case, although the time averaged x- and y-position errors are roughly of the same order, the 1-sigma deviation for the bending/vibration case tend to be .2 to .3 pixels greater than for the benchmark case in Table 6.3. This implies that the inclusion of the bending/vibration phenomena increases the uncertainty of the actual position

Table 6.6

Single Filter With Bending/Vibration;
CTR/SNG-S/T2/10-G/QB1

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$.4511 / .6175	.2088 / .7312
$\hat{y}_{err}(t_1^-)$.0098 / .6179	.4558 / .6266
$\hat{x}_{err}(t_1^+)$.3800 / .5877	.1636 / .7000
$\hat{y}_{err}(t_1^+)$.0109 / .5884	.4162 / .5872
$\hat{x}_{cent_{err}}(t_1^+)$.0022 / .1088	-.0129 / .1109
$\hat{y}_{cent_{err}}(t_1^+)$.0025 / .0647	.1584 / .0622
y peak-mean error (t_1^-) = -6.0 pixels		
y peak-mean error (t_1^+) = -4.0 pixels		
recovery time = 1.3 seconds		

of the target by 2/10 to 3/10 of a pixel, or, the impact of modeling the bending phenomena has a the same impact on the constant turn-rate tracker model as was evident on the Gauss-Markov model tracker. Again, as was the case for the Gauss-Markov model tracker, the time averaged statistics pertaining to the location of the centroid are not affected by the inclusion of the vibration and bending phenomena.

The effects of performing the simulation at the faster sampling time (50 Hz) and the effects due to modeling bending and vibration have shown comparable results to both the Gauss-Markov model tracker and the constant turn-rate model tracker (performance data not shown). Although the constant turn-rate model tracker has exhibited better performance than the Gauss-Markov model tracker at close ranges, the robustness issues to be addressed do not concern close range scenarios; therefore, subsequent simulations utilize the GM MMAF tracking algorithm since past studies have shown it to be preferable to the CTR model tracking algorithm at moderate to long ranges.

6.4 MMAF Bending/Vibration Analysis

An analysis comparable to that of the single filter case, Section 6.3, is performed with the GM MMAF tracking algorithm. In addition to the two scenarios analyzed in the single filter case (excluding, then including, the bending and vibration phenomena), two additional scenarios are considered. The four scenarios, which are distinguishable

by the magnitude of the zero frequency PSD value of bending noise (see Appendix A for the full development of K_b^2 , which is in fact that PSD value at zero frequency), are addressed in subsequent sections: 1) $K_b^2 = 0$, or effects of bending/vibration states are excluded; 2) $K_b^2 = 5 \times 10^{-13}$, includes the anticipated effect of bending/vibration without filter retuning; 3) $K_b^2 = 5 \times 10^{-12}$, increases the effect of bending/vibration states by an order of magnitude above the anticipated nominal values, again without filter retuning; and 4) same as (3) plus the addition of filter tuning. These analyses are performed with the Gauss-Markov, multiple-model adaptive filter tracking algorithm against a trajectory #2, 10-g maneuver.

6.4.1 MMAF Benchmark Simulation. Figures E-1 through E-8 show the benchmark of performance simulation for the case where the bending/vibration states are excluded from the truth model. These results are the same as those determined by Tobin (19:150-157). The statistics for the MMAF benchmark are shown in Table 6.7 and the dominant elemental filters are reflected in Table 6.8. These two sets of information serve as the basis of performance comparison for subsequent analyses involving the Gauss-Markov, Multiple Model Adaptive Filter, 10-G pull-up maneuver, trajectory #2 simulations.

Notice that the reacquisition algorithm discussed in Section 5.4 is accomplished at frames #74 and #75, as shown in Table 6.8. The state estimate of filters #1, #3, #4, and

Table 6.7

Multiple Model Adaptive Filter Benchmark;
GM/MMAF/T2/10-G

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.0024 / .4351	.2394 / .5155
$\hat{y}_{err}(t_i^-)$.0001 / .4327	.0968 / .7136
$\hat{x}_{err}(t_i^+)$.0116 / .3852	.2010 / .4303
$\hat{y}_{err}(t_i^+)$	-.0090 / .3950	.2311 / .6094
$\hat{x}_{-cent_{err}}(t_i^+)$	-.0033 / .1123	.0852 / .1787
$\hat{y}_{-cent_{err}}(t_i^+)$	-.0159 / .0634	.3991 / .2781
y peak-mean error (t_i^-) = -1.7 pixels		
y peak-mean error (t_i^+) = -.7 pixels		
recovery time = .6 seconds		

Table 6.8

Dominant Elemental Filters;
GM/MMAF/T2/10-G

Interval (Frames)	Dominant Filter(s)	Comments
1 - 65	#1	Tracking of benign trajectory; maneuver begins at frame 60
66 - 73	#2, #5	Y-direction maneuver recognized
74 - 75	#2	Other filters "lose lock"
76 - 120	#2, #4	Wide FOV tracking
121 - 150	#3	10-g maneuver recognized

#5 are all reset to match filter #2's state estimate since filter #2 is the only filter that doesn't "lose lock". At frame #74, a good estimate of the target's y-acceleration is predicted by filter #2, which is then transferred to the other filters during the reacquisition cycle. This transfer effectively resets the conditional mean value of the Gauss-Markov acceleration processes in the remaining filters, thus providing each filter an accurate estimate of actual y-direction target dynamics. It is reasonable, therefore, to stretch the FOV in the x-direction (weighting filter #4 appropriately) during this period since most of the changes in the target's acceleration now occur in the x-direction.

6.4.2 Effects of Bending/Vibration. This section addresses the impact of including the bending/vibration states in the simulation truth model but not telling the filter any additional information. Recall from Appendix A that the strength of the bending/vibration white noise, Q_b , is expressed in terms of the parameter K_b^2 and that the baseline value for K_b^2 is 5×10^{-13} (equal to the zero-frequency PSD value). Figures E-9 through E-16 show the performance plots for this simulation. Figures E-9 and E-10 show that the actual error exceeds the filter-computed error at random intervals throughout the simulation, thus implying that the filter is oblivious to the changes being contributed from the bending/vibration phenomena, as expected. As was the case for the single filter analysis, the MMAF appears to be more confident of its outputs than it should

Table 6.9

MMAF With Bending/Vibration;
GM/MMAF/T2/10-G/QB1

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.0410 / .6374	.1837 / .7821
$\hat{y}_{err}(t_i^-)$.0582 / .6463	.0944 / .7983
$\hat{x}_{err}(t_i^+)$.0498 / .5926	.1395 / .7224
$\hat{y}_{err}(t_i^+)$.0439 / .6076	.2306 / .6912
$\hat{x}_{-cent_{err}}(t_i^+)$.0010 / .1332	.0451 / .1471
$\hat{y}_{-cent_{err}}(t_i^+)$	-.0081 / .0718	.3143 / .2415
y peak-mean error (t_i^-) = -1.6 pixels		
y peak-mean error (t_i^+) = -.7 pixels		
recovery time = .6 seconds		

Table 6.10

Dominant Elemental Filter;
GM/MMAF/T2/10-G/QB1

Interval (Frames)	Dominant Filter(s)	Comments
1 - 66	#1	Tracking of benign trajectory; maneuver begins at frame 60
67 - 72	#2, #5	Y-direction maneuver recognized
73	#2	Other filters "lose lock"
74 - 138	#2, #4	Wide FOV tracking, intermittent attempts to lock on x-direction filter (#4)
139 - 150	#3, #4	10-g maneuver recognized, harsh changes in x-direction followed

be. Table 6.9 shows the time averaged statistics for this simulation. A comparison of the x- and y-direction error statistics of Table 6.9 to those of the benchmark case of Table 6.7 reveal that, although the estimate of the mean error in both cases, with and without bending/vibration, are very similar, the uncertainty about that mean is increased by an average of 0.2 pixels when the bending/vibration phenomena is introduced. This is a direct correspondence with that found in the single filter benchmark case. This increase in the uncertainty can also be seen by comparing Figures E-11 through E-14 to the benchmark equivalent plots of Figures E-3 through E-6. The estimate of the x- and y-centroid errors and their associated one sigma values for both cases show no apparent dissimilarities between the two, as was the case with the single filter analysis in Section 6.3. The dominant elemental filters for this simulation are shown in Table 6.10. As compared to the benchmark, in Table 6.8, the two cases reveal some differences. First of all, this tracking scenario tracks the benign portion of the simulation for one frame longer than did the benchmark, otherwise interpreted as an additional one frame delay before a maneuver was recognized. Secondly, and also surprisingly, the simulation "lost track" for only one frame (rather than two as did the benchmark) shortly after recognizing the y-jink maneuver. And thirdly, it takes the algorithm longer to recognize the 10-g maneuver when the bending phenomena is modeled.

6.4.3 Increased Bending/Vibration Phenomena. This section addresses the concern of what magnitude of filter performance degradation can be expected if the phenomena due to bending/vibration is an order of magnitude greater than expected, or $K_D^2 = 5 \times 10^{-12}$. Figures E-17 through E-24 show the performance data and Table 6.11 reflects the statistics for this simulation. Figures E-17 and E-18 reveal the adverse impact as a result of the increased effect of the bending/vibration phenomena. Although the actual RMS error has increased significantly, the filter-computed RMS error ignores these effects. This is comparable to the case of a finely-tuned filter for a scenario which is really not present, which is expected since the bending phenomena is not modeled in the filter. The temporally averaged statistics of Table 6.11 show a significant difference in the one sigma values about the mean x- and y-error parameters, although the mean-error estimates had not changed significantly. The standard deviation of the errors has increased by a factor of four to five times beyond what was seen in the benchmark simulation. This wide spread of the one sigma error is clearly visible in Figures E-19 through E-22. Since the average uncertainty associated with the x- and y-error parameters lie in the range from 1.5 to 2.0 pixels, filter tuning is appropriate in an attempt to force the filter to "recognize" the changing, dynamic environment which it is ignoring, again,

Table 6.11

MMAF with Increased Bending/Vibration
GM/MMAF/T2/10-3/QB2

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$.0927 / 1.539	.1294 / 1.976
$\hat{y}_{err}(t_1^-)$.1610 / 1.515	.2304 / 1.625
$\hat{x}_{err}(t_1^+)$.1004 / 1.915	.0741 / 1.915
$\hat{y}_{err}(t_1^+)$.1397 / 1.449	.3446 / 1.528
$\hat{x}_{-cent_{err}}(t_1^+)$	-.0345 / .1493	.0053 / .1570
$\hat{y}_{-cent_{err}}(t_1^+)$.0062 / .0932	.1862 / .1976
y peak-mean error (t_1^-) = -1.7 pixels		
y peak-mean error (t_1^+) = -.7 pixels		
recovery time = .6 seconds		

which is expected due to the original tuning. Even though the filter grossly underestimates the error variances in its estimates of the target's x- and y-position, the x- and y-centroid errors do not appear to be affected as compared to the benchmark simulation. This shows that, although the state estimates are corrupted via the reduced-order filter model, there is little impact on centering the image for averaging.

6.4.4 Increased Bending/Vibration With Tuning. This section addresses the attainable performance when the order of magnitude of bending/vibration phenomena such as shown in

Section 6.4.3 is evident in the truth model but the filter is allowed to be retuned for best estimation accuracy. In selecting the appropriate filter tuning parameters, the filter "Q" is disregarded since the dynamics models have not varied from the baseline simulation. The filter "R" value, however, should vary due to the fact that the bending and vibration phenomena was not modeled in the filter (recall Equations (6-2) and (6-3)) and, in fact, the "R" is expected to increase, to account for the variance (RMS) contribution of x_b and y_b to z_1 and z_2 , respectively. An initial approximation as to what "R" value to choose can be computed from the adaptive estimation relation (10:122):

$$\underline{R}(t_i) = \frac{1}{N} \sum_{j=i-N+1}^i [\underline{x}_j \underline{x}_j^T - \underline{H}(t_j) \underline{P}(t_j^-) \underline{H}^T(t_j)] \quad (6-2)$$

The first approximation of "R" revealed that it should be increased almost two orders of magnitude greater than the empirically determined values identified in Equation (4-22) without bending effects considered. This increase, however, only improved tracking performance slightly better than what had been shown without the additional tuning. A closer examination of the mean-squared value contribution associated with the bending states (over a single Monte Carlo simulation) revealed that an increase of four orders of magnitude, rather than two, over the baseline "R" values was appropriate. Filter tuning for the "best" values resulted in the following, which is the result of adding $4\underline{I}$ to the

values identified in Equation (4-22) (the value of 4 was derived from taking the average contribution of the variance from x_D and y_D over a single Monte Carlo run):

$$\underline{R} = \begin{bmatrix} 4.00436 & 0 \\ 0 & 4.00598 \end{bmatrix} \quad (6-3)$$

The corresponding performance plots are shown in Figures E-25 through E-32 and the statistical data appears in Table 6.12. Notice that the filter exhibits a delayed response to the maneuver by approximately one-third of a second as illustrated by the filter-computed RMS error curves of Figures E-25 and E-26. In addition, these plots show the effect of the increased measurement noise "masking" the filter-computed RMS values, implying that the filter is tuned to place less trust in the measurements and to rely more on the internal filter dynamics models. The increase in measurement noise also implies that there lies a greater uncertainty associated with the measurements, which is indicated in Figures E-27 through E-30 by the larger standard deviation as compared to the previous case without the additional tuning, as depicted in Figures E-18 through E-24. This relationship can also be seen by comparing the x- and y-statistics shown in Table 6.12 with the corresponding data in Table 6.11. In addition to the increased standard deviation, the y-channel also displays a bias of approximately -4.0 pixels, beginning at the onset of the maneuver and

Table 6.12

MMAF with Increased Bending and Tuned
GM/MMAF/T2/10-G/QB2/R+4

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	{0.5 , 2.0}	{3.5 , 5.0}
$\hat{x}_{err}(t_1^-)$	-.0660 / 1.515	.5191 / 2.102
$\hat{y}_{err}(t_1^-)$.0200 / 1.509	-1.392 / 1.581
$\hat{x}_{err}(t_1^+)$	-.0506 / 1.462	.4398 / 2.049
$\hat{y}_{err}(t_1^+)$.0136 / 1.456	-3.703 / 1.514
$\hat{x}_{cent_{err}}(t_1^+)$	-.1880 / .6455	.4274 / .5502
$\hat{y}_{cent_{err}}(t_1^+)$	-.0932 / .5986	-4.015 / .4384
y peak-mean error (t_1^-) = -6.5 pixels		
y peak-mean error (t_1^+) = -4.5 pixels		
recovery time = .6 seconds		

continuing throughout the duration of the simulation. This phenomenon is also exhibited by the estimated y-centroid error as illustrated in Figure E-32, with an initial bias of about -3.0 pixels and increasing (negatively) in time. These biases in the y-channel are a result of the increased measurement variances in the filter-assumed model. Examination of the dominant elemental filters for this simulation revealed that filter #2, the large FOV filter, remained dominant throughout after the time of the maneuver initiation. This implies that the increased measurement variance inhibited the ability of the MMAF algorithm to

distinguish between the "good" and "bad" filters, and thus the increased bias in the y-channel is due to filter #2's poor estimate of y-channel dynamics. This can also explain the exceptionally high time-averaged y-centroid error parameter value shown in Table 6.12 between 3.5 and 5.0 seconds. As a result of the proposed tuning method, Figures E-24 and E-25 indicate that better performance is attained with the higher "R" values; however, Figures E-26 through E-32 indicate the converse (especially in the y-channel).

As a further investigation into this concept of the unexpected bias contribution, the discrete-time measurement noise matrix was changed to reflect the values of 1.5 where the square root of this is the standard deviation contribution from the bending states over a ten run Monte Carlo simulation (it makes more sense to alter the measurement matrix by the variance; however, the intent of this simulation is to show a trend, rather than a physical realizability). The performance plots for this simulation are shown in Figures E-33 through E-40. Figures E-33 and E-34 reflect that "better" tracking as compared to the case without tuning is effected; however, a similar occurrence to that of the previous tuned case is evident - a biasing effect appears after the maneuver initiation in the y-channel. Again, the dominant filter after the maneuver was filter #2, and, thus, this shows that too large of a measurement variance was being utilized in order for the algorithm to adequately distinguish the correct model.

6.4.5. Impact of Bending/Vibration Phenomena. As has been shown, the effect of modeling bending and vibration in the truth model but not providing the filter such additional information does not impede the Gauss-Markov MMAF tracking algorithm from accurately pointing and tracking, provided the level of bending and vibration is on the order of that expected. However, if the level of bending and vibration is greater than that which is expected, tracking can be maintained but accurate pointing for the purpose of focusing a weapon such as a laser is severely degraded. The two tuning exercises associated with the increased bending level have implied that the result of increasing the measurement variance to account for the unmodeled effects impedes the MMAF algorithm's ability to choose the appropriate filter model in the bank.

6.5 Scenario Sensitivity Analysis

This section addresses the sensitivity of the tracking algorithm to variations of pixel size, target type, target range, and noise characteristics. The simulations performed for this analysis do not include the effects of bending and vibration in the truth model; in addition, software modifications have been introduced at this point to accommodate the description of two additional trajectories which are introduced in the following section. These simulations are performed using the Gauss-Markov, MMAF

tracking algorithm employing a trajectory #2 configuration. Table 6.13 shows the scenarios considered for this analysis.

The target type parameters, or correlation time pairs, of 8/3, 4/1.5, 4/.8, and 4/.4 are perceived to represent the dynamics of a long-range missile, a bomber aircraft, a high-performance fighter aircraft, and an air-to-air missile, respectively. These time pairs identify the correlation times assumed by the filter for both the benign and the dynamic case; for the benchmark, the benign $\tau = 4$ seconds and the dynamic $\tau = 1.5$ seconds. The pixel size corresponds to the angular field-of-view of a single pixel and the range parameter corresponds to the target's initial condition at a location perpendicular to the FLIR FOV. With respect to pixel size, those simulations that vary from the nominal (nominal = 20 μ rad on a side) necessitate changing the inertial x- and y-initial conditions in order to provide the tracker the identical initial angular orientation of the target as seen by the FLIR FOV as in previous studies. This relationship with respect to elevation and azimuth is depicted in Figure 2.6.

The noise parameters, Q_{R1} and Q_{R2} , are the respective continuous-time noise strengths (nominal values determined by computing Equation (5-1) on the values depicted in Table 5.2) for the dynamics driving noises for the models upon which the filters are based. Recall from Equation (5-1) that "Q" is expressed as a function of the acceleration variance (or mean squared value) and correlation time for

Table 6.13
Scenario Sensitivity Parameters

#	Target Type (seconds)	Pixel Size (radians)	Range (meters)	Noise (Q)	Maneuver
1.	8/3	200 nrad	2000000	Q _{R1}	2-G
2.	8/3	200 nrad	2000000	Q _{R1}	10-G
3.	4/1.5	20 μ rad	20000	Q _{R1}	10-G
4.	4/1.5	20 μ rad	20000	Q _{R1}	20-G
5.	4/.8	20 μ rad	20000	Q _{R1}	10-G
6.	4/.8	20 μ rad	20000	Q _{R1}	20-G
7.	4/.4	20 μ rad	20000	Q _{R1}	10-G
8.	4/.4	20 μ rad	20000	Q _{R1}	20-G
9.	4/.4	2 mrad	20000	Q _{R2}	10-G
10.	4/.4	2 mrad	20000	Q _{R2}	20-G

the first-order Gauss-Markov filters. The parameter Q_{R1} in Table 6.12 implies that, for those simulations, the zero frequency PSD (and low frequency) value for "Q" must be equivalent to that of the benchmark simulation in order to retain a viable performance analysis comparison, or, the mean squared value of acceleration (where the energy corresponds to the area under the PSD curve of Figure 6.1) remains the same between the scenarios to be compared. Therefore, for those simulations where the target type is different from that of the benchmark simulation ($\tau = 4/1.5$ seconds), the corresponding filter variance must also be adjusted to

retain this relationship. Figure 6.1 illustrates the relationship between "Q" and the PSD curve for a first-order Gauss-Markov process. The noise parameter Q_{R2} is an exception to the above discussion based upon the intuitive notion that, as the pixel size decreases in magnitude without a corresponding increase in range magnitude, then a similar proportional increase of the filter's driving noise strength must occur in order to compensate for the "apparent" changes in target dynamics due to decreased pixel size. Note that adjusting the baseline "Q" value, Q_{R1} , is not necessary for the first scenario listed in Table 6.13 as compared to the nominal since the pixel size not only decreased by two orders of magnitude from the nominal but the initial (nominal = 20,000 meters) range coordinate also increased by

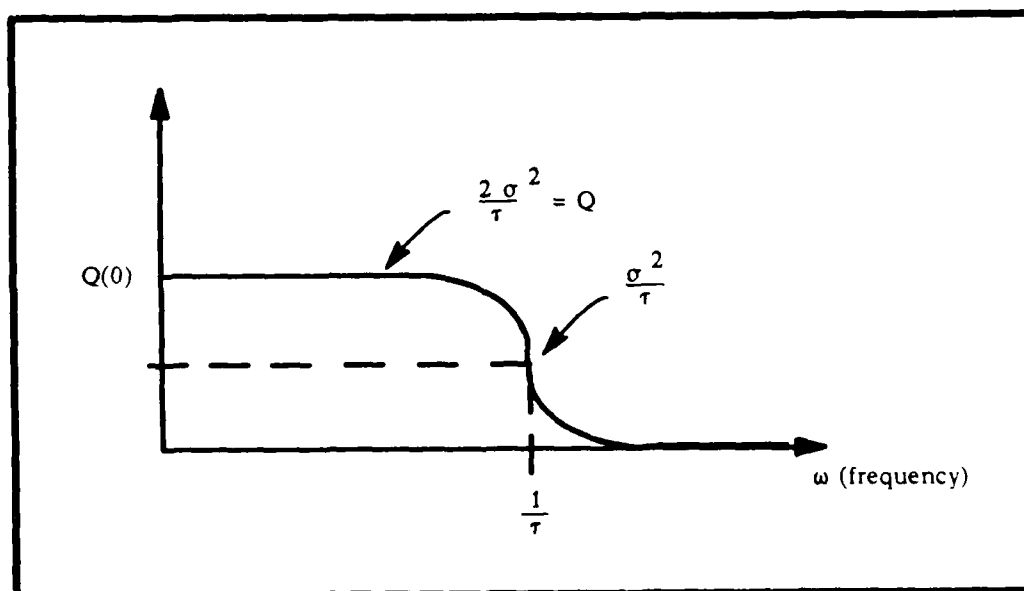


Figure 6.1 PSD Plot of a First-Order Gauss-Markov Process

two orders of magnitude. This corresponds to the argument that, as the pixel size decreases/increases in magnitude and the range to target increases/decreases by a proportional degree of magnitude, the driving noise strength necessary in the filter to depict the same magnitude of target maneuvers is unchanged. In addition, this differs from Tobin's "Q-vs-Range" function (19:126) described by:

$$Q_{r2} = (r_2/r_1)^2 Q_{r1} \quad (6-4)$$

where:

Q_{r1} = strength of driving noise for a target at range r_1

Q_{r2} = strength of driving noise for a target at range r_2

where this relationship is based upon targets at different ranges while maintaining the equivalent pixel size.

Rather than discussing all ten of the simulation scenarios listed in Table 6.13, scenarios 1, 4, 5, 8, and 9 will be discussed since these simulations encompass the different variations of parameter values. Scenario #2 is the same as #1 but with the turn maneuver occurring at a higher g-level. Scenario #3 is the same as the benchmark run discussed in Section 6.4.1 and scenarios #6, #7, and #10 are similar to #5, #8, and #9, respectively, with the exception of the different g-level maneuver.

6.5.1 Range/Pixel Size Sensitivity. Scenario #1 considers the case where the pixel size is decreased by two orders of magnitude and the range-to-target is increased by two orders of magnitude (nominal range = 20000 meters).

This scenario is performed for a target represented by a "benign" correlation time of 8 seconds and a "dynamic" correlation time of 3 seconds throughout the 2-g turn. The performance plots for this simulation are found in Figures F-1 through F-8 and the time averaged statistics are compiled in Table 6.14. The statistics for this simulation are very similar to those of the benchmark case as depicted in Table 6.7 with one unique exception: the y-peak-mean error is approximately 1/2 that of the benchmark run, or the filter does a better job estimating the target's position at the time immediately following the harsh maneuver when the

Table 6.14

MMAF Scenario #1 Statistics
GM/MMAF/T2/2-G/T=8, 3/2₀=2000000

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.1572 / .5273	.1141 / .5398
$\hat{y}_{err}(t_i^-)$	-.0258 / .3543	-.0426 / .3760
$\hat{x}_{err}(t_i^+)$.1360 / .4230	.1156 / .4331
$\hat{y}_{err}(t_i^+)$	-.0223 / .3114	-.0259 / .3354
$\hat{x}_{-cent_{err}}(t_i^+)$.1223 / .1035	.0566 / .1345
$\hat{y}_{-cent_{err}}(t_i^+)$.0049 / .0589	-.0043 / .0817
y peak-mean error (t_i^-) = -.9 pixels		
y peak-mean error (t_i^+) = -.6 pixels		
recovery time = .7 seconds		

target is positioned at great distances from the sensor. Figure F-2 shows this phenomenon, as compared to the benchmark case in Figure E-2, where the actual RMS error changes drastically for the time between 2.0 and 3.0 seconds and the filter does a good job of following the changes. Also of particular interest are the different characteristics associated with the estimated x- and y-centroid positions as shown in Figures F-7 and F-8 as compared to Figures E-7 and E-8 (note the difference in scales). The estimated x-plus centroid position exhibits a positive bias, or overestimate, until the time when the maneuver is initiated, shown in Figure F-7, as compared to the relatively stable estimation shown in Figure E-7. On the contrary, the estimated y-plus centroid position exhibits a much more stable estimation, shown in Figure F-8, as compared to that of Figure E-8, which exhibits a bias for time greater than 3.5 seconds.

6.5.2 High-g Maneuver Sensitivity. Scenario #4 represents a target of correlation time pair of 4/1.5 seconds pulling a 20-g maneuver at nominal range and pixel size and utilizing the baseline values of Q . Performance plots for this simulation are shown in Figures F-9 through F-16 and the statistical data is reflected in Table 6.15. As had been demonstrated in a previous research effort by Tobin (19:107,109), the filter performed rather well against this scenario, which serves as the benchmark to measure the effectiveness of subsequent 20-g scenario simulations. Note how fast the filter recovers from the maneuver, as can be

Table 6.15

MMAF Scenario #4 Statistics
GM/MMAF/T2/20-G/T=4,1.5

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$	-.0456 / .4261	.5565 / .5519
$\hat{y}_{err}(t_i^-)$	-.0105 / .3577	.4192 / .6163
$\hat{x}_{err}(t_i^+)$	-.0354 / .3707	.4294 / .4377
$\hat{y}_{err}(t_i^+)$	-.0137 / .3173	.5792 / .5144
$\hat{x}_{-centerr}(t_i^+)$	-.0006 / .0948	.2694 / .1697
$\hat{y}_{-centerr}(t_i^+)$	-.0015 / .0536	.6919 / .2719
y peak-mean error (t_i^-) = -2.3 pixels		
y peak-mean error (t_i^+) = -1.1 pixels		
recovery time = .5 seconds		

seen by comparing Figure F-2 to E-2, thus implying that the filters are "better" tuned for 20-g maneuvers than they are for 10-g maneuvers. This is also evident in Figures F-6 and E-6 in the estimation of the y-plus position and Figures F-8 and E-8 in estimating the y-plus centroid position.

6.5.3 Medium Correlation Time Target. Scenario #5 represents a target of correlation time pair 4/.8 seconds at nominal pixel size, range, and noise strength pulling a 10-G turn. The variances associated with elemental filters #2, #3, #4 x-dynamics, and #5 y-dynamics were adjusted to 5333, 1066, 5033, and 5033, respectively. This adjustment retains

the equivalent mean squared value of acceleration that was available to the filters in the benchmark runs so that a one-to-one comparison is effected (pictorially illustrated in Figure 6.1). The performance plots for this simulation appear in Figures F-17 through F-24 and the statistics are shown in Table 6.16. Evident in both the statistical data and the performance plots are characteristics comparable in performance to the benchmark, with the exception that the x-channel estimates shown in Figures F-19, F-21, and F-23 (as compared to Figures E-3, E-5, and E-7, also note difference in scales) for both position and centroid position exhibit a time-increasing bias after roughly 0.5 seconds after the

Table 6.16

MMAF Scenario #5 Statistics
GM/MMAF/T2/10-G/ τ =4,.8

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$	-.0516 / .4243	.2955 / .5216
$\hat{y}_{err}(t_i^-)$	-.0097 / .3574	-.0735 / .6532
$\hat{x}_{err}(t_i^+)$	-.0374 / .3699	.2476 / .4196
$\hat{y}_{err}(t_i^+)$	-.0125 / .3174	.0912 / .5883
$\hat{x-cent}_{err}(t_i^+)$.0010 / .0978	.1419 / .1456
$\hat{y-cent}_{err}(t_i^+)$	-.0005 / .0541	.3575 / .3041
y peak-mean error (t_i^-) = -1.7 pixels		
y peak-mean error (t_i^+) = -.7 pixels		
recovery time = .5 seconds		

maneuver has been initiated. The probable cause of the bias is the fact that the filters were previously tuned for a target correlation time of 1.5 seconds, thus, the strength of the driving white noise, "Q", was predicated on this type of correlation time. This is comparable to optimally tuning the Kalman filters to track bomber type aircraft but performing the tracking simulation against fighter type aircraft. One means to handle this phenomenon would be to provide adaptive estimation of parameters such as the filter "Q", or other parameters where appropriate, and adjust the parameters to react to the changing environment.

6.5.4 Fast Correlation Time Target. Scenario #8 represents a target with correlation time pair of 4/.4 seconds at nominal pixel size, range, and noise strength pulling a 20-G turn. Again, the variances related with elemental filters #2, #3, #4 x-dynamics, and #5 y-dynamics were adjusted to 2667, 533, 2667, and 2667, respectively, to account for the matching of the mean squared value of acceleration between this simulation and the benchmark (see Figure 6.1). For performance comparison purposes, this simulation is measured against the performance exhibited in the Scenario #4, the 20-g benchmark. The performance plots for this simulation appear in Figures F-25 through F-32 and the statistical data is shown in Table 6.17. In comparing the performance data, Figures F-27, F-29 and F-31, as compared to Figures F-11, F-13, and F-15 (note difference

Table 6.17

MMAF Scenario #8 Statistics
GM/MMAF/T2/20-G/T=4,.4

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$	-.0679 / .4213	1.175 / .5201
$\hat{y}_{err}(t_1^-)$	-.0089 / .3557	-.8953 / .3924
$\hat{x}_{err}(t_1^+)$	-.0467 / .3702	.8846 / .4199
$\hat{y}_{err}(t_1^+)$	-.0119 / .3154	-.6203 / .3609
$\hat{x}_{cent_{err}}(t_1^+)$.0013 / .0970	.5576 / .1525
$\hat{y}_{cent_{err}}(t_1^+)$.0004 / .0528	.0665 / .1045
y peak-mean error (t_1^-) = -3.0 pixels		
y peak-mean error (t_1^+) = -1.5 pixels		
recovery time = .7 seconds		

in scales), the bias in the x-channel as seen in the scenario #5 is prevalent. In fact, this bias is larger than seen previously, primarily due to the greater mismatch of the correlation times as compared to the benchmark. Again, a parameter estimation technique would seem appropriate to compensate for the large bias errors.

6.5.5 Pixel Size/Noise Strength Relationship.

Scenario #9 represents a target of correlation time pair of 4/.4 seconds at nominal range and pixel size of two milliradians pulling a 10-G maneuver. The noise strength for this simulation was decreased by two orders of magnitude for each filter to correspond to the same increased pixel

size from nominal. The performance plots appear in Figures F-33 through F-40 and the statistical data is depicted in Table 6.18. Of particular interest is the high x-position estimation error for time less than 1.2 seconds as compared to the benchmark simulation. Although this is evident in Figure F-33, it is more clearly visible in the mean error estimate shown in Figures F-35 and F-37. The mean value of the error in the filter's prediction estimate has increased by almost two orders of magnitude compared to what had been experienced in the benchmark scenario. This is felt to be due to the fact that, although the pixel size had increased

Table 6.18

MMAF Scenario #9 Statistics
GM/MMAF/T2/10-G/T=4,.4/KPT=.002

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_1^-)$.2263 / .4067	.0753 / .3103
$\hat{y}_{err}(t_1^-)$	-.0440 / .3241	-.0932 / .3087
$\hat{x}_{err}(t_1^+)$.1616 / .2964	.0756 / .2916
$\hat{y}_{err}(t_1^+)$	-.0422 / .2964	-.0761 / .2889
$\hat{x}_{cent_{err}}(t_1^+)$.0070 / .1021	.0132 / .0982
$\hat{y}_{cent_{err}}(t_1^+)$	-.0156 / .0634	-.0160 / .0632
y peak-mean error (t_1^-) = -.3 pixels		
y peak-mean error (t_1^+) = -.2 pixels		
recovery time = 1.1 seconds		

by two orders of magnitude, the initial position and velocity parameters associated with the target truth model had not changed; therefore, the target "appears" to be moving less harshly at the onset of the simulation to the larger pixel size than it had for the nominal case. Another interesting facet about this simulation is the stability associated with tracking the estimated y-position. The peak-mean error estimate after the onset of the turning maneuver had improved to 1/6th of that estimated by the benchmark run given in Table 6.7 (as expressed in pixels, even though the physical pixel sizes themselves have changed between the two cases).

6.6 Target Trajectory Sensitivity

This section analyzes the sensitivity of the tracking algorithm against two target trajectories, #5 and #6, for which the algorithm had not been previously tuned. Unlike the sensitivity studies in the last section, the effects of bending/vibration are included in the truth model for these analyses. The trajectory sensitivity analyses are conducted with the GM MMAF tracking algorithm against targets pulling a 10-g turn.

6.6.1 Sensitivity to Trajectory #5. The first trajectory to be analyzed is trajectory #5. As shown in Figure 2.5, trajectory #5 is accomplished by rotating trajectory #2 by 45° , thus reorienting the truth model trajectory generation in space rather than redefining the

orientation of the inertial coordinate system. The performance plots are shown in Figures G-1 through G-8, the statistical data appears in Table 6.19, and the dominant filters throughout the tracking simulation are depicted in Table 6.20. The statistical data shows that both the time averaged mean and one sigma deviations have increased for all time as compared to the benchmark. However, a direct comparison of x-channel or y-channel results for the two cases doesn't make total sense, since the trajectory characteristics are now rotated with respect to the tracker's x- and y-directions. Also, reviewing the time histories of y-minus position and y-plus position show that the filter initially (for time less than .8 seconds), has difficulty in estimating the actual y-position, which is depicted by the mean error estimates in Figures G-3 and G-5. This phenomena had only been experienced at the onset of the turn maneuver in previous scenario simulations. In addition, the data enumerated in Table 6.20 shows that the MMAF algorithm does not consistently identify a dominant filter for any one significant stage of the scenario. This is due to the orientation of the target and the orientation of the velocity vector not dominating any one direction with respect to the field-of-view, especially prior to the maneuver initiation, but equally represented in both the FLIR x- and y-directions.

Table 6.19

Trajectory #5 Statistical Data
GM/MMAF/T5/10-G

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.0770 / .6824	.1464 / .8269
$\hat{y}_{err}(t_i^-)$.0407 / .6911	.1999 / .6552
$\hat{x}_{err}(t_i^+)$.0730 / .6302	.0565 / .7491
$\hat{y}_{err}(t_i^+)$.0534 / .6106	.2284 / .5897
$\hat{x}_{-cent_{err}}(t_i^+)$	-.0083 / .1048	-.0748 / .1238
$\hat{y}_{-cent_{err}}(t_i^+)$.0420 / .0802	.1353 / .1789
y peak-mean error (t_i^-) = -1.4 pixels		
y peak-mean error (t_i^+) = -1.1 pixels		
recovery time = .6 seconds		

Table 6.20

Trajectory #5 Dominant Filters

Frame	Dominant Filter(s)	Remarks
1 - 14	#1	Acquisition phase
15 - 16	#1, #5	Attempts to follow a y-direction maneuver
17 - 69	#1, #2, #3, #5	Intermittent behavior between y-direction and 10-g maneuver
70 - 150	#2, #4	Attempts to follow a x-direction maneuver

6.6.2 Ramping Acceleration at Turn Initiation.

Trajectory #6 is of the same configuration as trajectory #2 with the exception that when the pull-up maneuver occurs (time = 2.0 seconds), the acceleration increases as a ramp function and attains its maximum after 5 sample periods have expired, rather than occurring as a step change. This type of acceleration profile is more realistic, less harsh than the artificial step increase in acceleration. Figure 6.2 shows the initial acceleration profiles at commencement of the pull-up maneuver for both trajectory #2 and #6. Notice that the greatest difference between the two trajectories occurs in the x-direction acceleration profile. The trajectory #2 accelerations for the x- and y-directions are expressed as:

$$a_x = -\omega V_{\text{mag}} \cos(\omega t) \quad (6-3)$$

$$a_y = \omega V_{\text{mag}} \sin(\omega t) \quad (6-4)$$

where:

V_{mag} = magnitude of velocity at $t=0$

ω = turn-rate of the target

t = time, measured from maneuver initiation
= $t_{\text{simulation}} - 2 \text{ sec}$ (Note: applicable for the maneuver to begin at $t_{\text{simulation}} = 2.0 \text{ sec}$)

To effect a ramping acceleration, Equations (6-3) and (6-4) are premultiplied by a dimensionless ramp function of the following form for the first 5 sample periods after maneuver initiation:

$$r(t) = 6t \quad (6-5)$$

where the 6 is present to account for the ramping effect to attain its maximum after 5 sample periods (each sample is 1/30th of a second). Thus, multiplying Equations (6-3) and (6-4) by Equation (6-5) and integrating the outcome produces

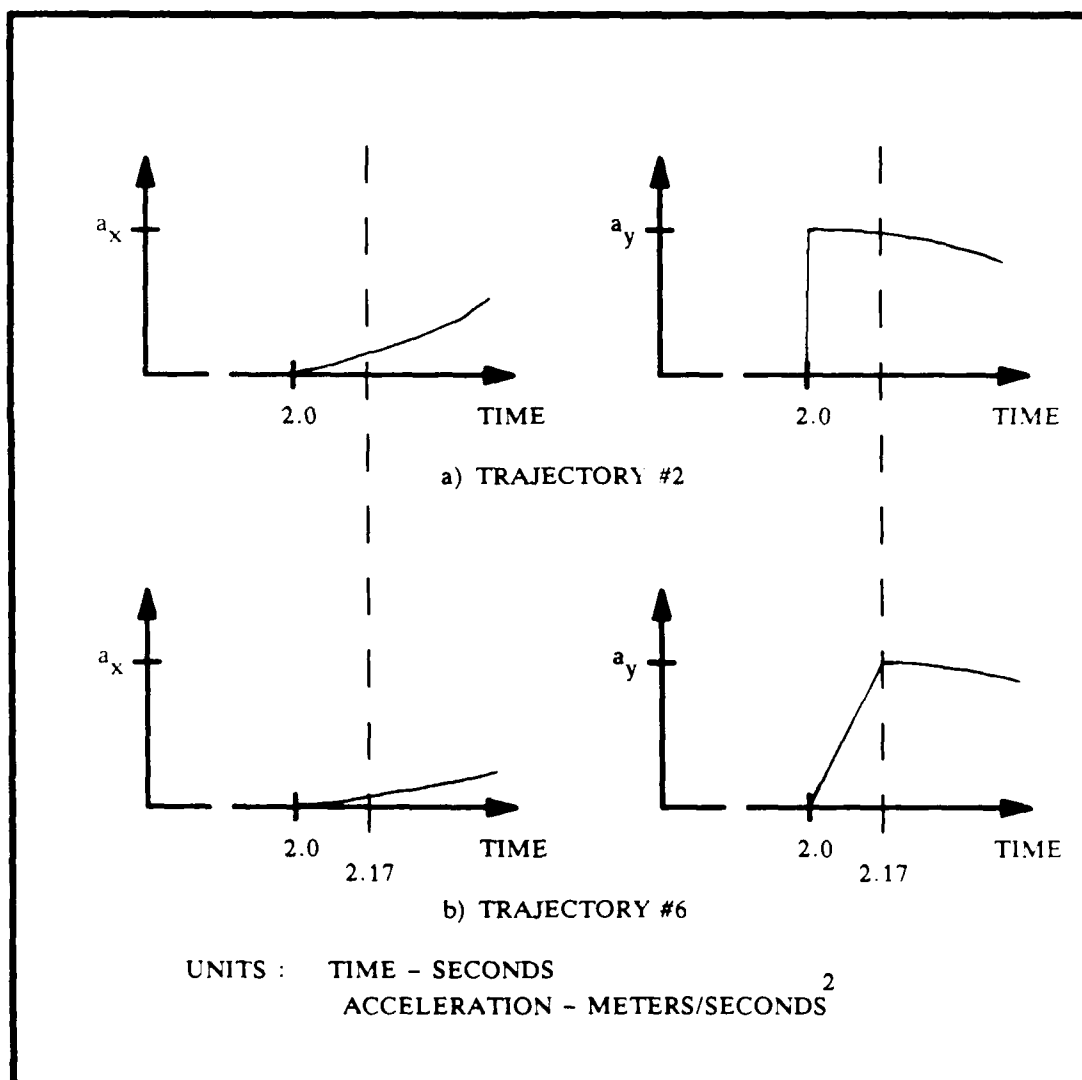


Figure 6.2 Trajectory #2 and #6 Acceleration Profiles

the resultant x- and y-direction inertial velocity relationships described by:

$$V_x = V_{mag}\{6[t\cos(\omega t) - (1/\omega)\sin(\omega t)] - 1\} \quad (6-6)$$

$$V_y = 6V_{mag}\{(1/\omega)[\cos(\omega t) - 1] + t\sin(\omega t)\} \quad (6-7)$$

and

$$V_x = V_{mag} \text{ @ } t = 0.0$$

$$V_y = 0.0 \text{ @ } t = 0.0$$

Integrating Equations (6-6) and (6-7) provide the following x- and y-direction position relationships:

$$x_t = 6V_{mag}\{2/\omega^2[\cos(\omega t) - 1] + t/\omega\sin(\omega t) - t\} + x_{t0} \quad (6-8)$$

$$y_t = 6V_{mag}\{[2/\omega^2\sin(\omega t)] - t/\omega[1 + \cos(\omega t)]\} + y_{t0} \quad (6-9)$$

where:

$$x_{t0} = \text{true x-position at time} = 0.0$$

$$y_{t0} = \text{true y-position at time} = 0.0$$

Upon implementing these relationships into the GM MMAF tracking algorithm, a slight enhancement in y-direction performance is expected for the simulation beyond time = 2.0 seconds as compared to the benchmark data with bending included appearing in Figures E-9 through E-16. The performance plots appear in Figures G-9 through G-16 and the statistical data appears in Table 6.21. Upon comparing the performance plots for the two cases, no apparent differences are discernible. As the data in Table 6.21 reveals, the mean errors in the estimates of y-position and y-centroid error have improved both at time equal to t_1^- and t_1^+ as

compared to the benchmark including bending and vibration as shown in Table 6.9. In addition, when comparing the one sigma values for all time, this simulation (for which the acceleration begins as a ramp) shows a slight improvement over the case where the acceleration acts like a step in the y-direction, which is expected. One additional note is that the dominant elemental filters for this simulation (not shown) are basically the same as Table 6.10.

Table 6.21
Trajectory #6 Statistical Data
GM/MMAF/T6/10-G/QB1

Temporally Averaged Error Parameter (mean / 1 sigma)	Time Interval	
	[0.5 , 2.0]	[3.5 , 5.0]
$\hat{x}_{err}(t_i^-)$.0427 / .6309	.1878 / .7882
$\hat{y}_{err}(t_i^-)$.0586 / .6439	.0397 / .7693
$\hat{x}_{err}(t_i^+)$.0510 / .5869	.1431 / .7275
$\hat{y}_{err}(t_i^+)$.0439 / .6052	.1803 / .6683
$\hat{x}_{center}(t_i^+)$.0025 / .1318	.0468 / .1463
$\hat{y}_{center}(t_i^+)$	-.0091 / .0706	.2814 / .2204
y peak-mean error (t_i^-) = -2.2 pixels		
y peak-mean error (t_i^+) = -.8 pixels		
recovery time = .6 seconds		

6.7 Compendium.

This chapter has analyzed several aspects pertaining to airborne tracking systems simulations. First, the potential of operating at a 50 Hz sample frequency versus a 30 Hz rate was investigated for both the Gauss-Markov tracker and the constant turn-rate model tracker, utilizing a single filter configuration. Secondly, both tracker models were evaluated with respect to adding the bending and vibration effects of a large space structure to the truth model without changing the basic structure of the filter, thus representing a reduced-order filter. After evaluating these effects against the single filter configuration, the same analysis was performed with the Gauss-Markov MMAF tracking algorithm. In addition, the magnitude of the bending was increased and filter tuning was performed to compensate for this unmodeled effect. After the bending and vibration analysis, a sensitivity study was performed in order to evaluate the effects of changing pixel size, target type, or correlation time, driving noise strength, and sensor-to-target range. This scenario sensitivity study was then followed by a sensitivity study of two target trajectory profiles for which the filters had not been tuned. The conclusions drawn from these analyses are summarized in the following chapter.

VII. Assessments and Recommendations

7.1 Introduction

This chapter assesses the impact of the simulation analyses discussed in the previous chapter and suggests research areas requiring further study. Section 7.2 examines the simulations discussed in Chapter VI and evaluates each for their effect on the FLIR tracking algorithms. Section 7.3 addresses the preliminary studies performed on investigating a rotating rectangular field-of-view concept and Section 7.4 enumerates recommendations for further study.

7.2 Assessments

7.2.1 Practicality of the 50 Hz Sample Frequency. The potential of implementing the tracking algorithm at a 50 Hz rate rather than at a 30 Hz rate was investigated. The impact of processing at the 50 Hz rate results in slight improvements in estimation: both the mean error and the variance about the mean error in both the x- and y-directions decrease by a small amount (average 6% decrease). However, the computer loading penalty paid as a result of the more frequent sampling greatly outweighs the potential benefits obtainable. As an example, the computer processing time associated with a ten Monte Carlo simulation utilizing the single filter Gauss-Markov tracking model results in a 73% increase for the 50 Hz version over that which had been experienced with the 30 Hz version. One would expect a 67%

increase since $50/30 = 1.67$; the remainder of the increase is due, perhaps, to the overhead associated with the statistical evaluations when performed on the CYBER. As a consequence, the 30 Hz sampling frequency is sufficient for the tracking algorithm to provide good performance and at the same time minimize computer processing time.

7.2.2 Impact of Hardware Bending/Vibration. The issue related to the repercussions due to the bending/vibration phenomena addresses two concerns. The first concern considers the effect on the filter algorithm as the expected degree of bending and vibration of the space structure (see Appendix A) is considered in the truth model while the filter is not provided information regarding this phenomena. This is a robustness concern.

Sections 6.3.1 and 6.3.2 discuss this impact with respect to single filter tracking algorithms based on Gauss-Markov acceleration and constant turn rate models, respectively, and Section 6.4.2 discusses this issue with respect to Gauss-Markov MMAF algorithm. In all three instances, the inclusion of the bending and vibration phenomena displays little or no effect to the mean position errors as compared to the same scenario without the bending/vibration states. However, the error standard deviations associated with the error reflected increases on the order of 0.2 to 0.3 pixels. Only in the case of the y-position estimate mean error and one standard deviation with

the CTR filter did the total RMS error exceed one pixel, thus distorting the tracking ability of the algorithm. Both Gauss-Markov algorithms, the single filter and the MMAF, maintained favorable tracking ability with the expected vibration and bending levels included in the truth model.

The second concern addresses the effect of increasing the vibration/bending phenomena by an order of magnitude; this was presented in Section 6.4.3 without additional tuning. This simulation showed that, although the filter's mean errors did not change significantly, the one sigma values increased by a factor of four or five, to between 1.5 to almost 2.0 pixels. The desired tracking accuracy to less than a pixel in resolution is severely impeded by the large standard deviation values. Thus, to maintain adequate resolution and tracking performance capability, the degree of bending and vibration due to the flexure of the structure should not exceed that discussed in Appendix A, provided no filter remodeling has compensated for this effect. If retuning is allowed, as was presented in Section 6.4.4, it was shown that the MMAF's ability to select the appropriate filter model for accurate tracking and pointing was severely impeded by the measurement variance compensation.

7.2.3 Filter Sensitivity to Varying Parameters. This section summarizes the relationships observed for varying selected parameters in the tracking simulation as discussed in Section 6.5. The first scenario, scenario #1, shows that increasing target range by two orders of magnitude from

nominal, where the nominal is 20,000 meters, results in comparable performance (mean \pm 1 sigma values) to the benchmark scenario when the pixel size is decreased by two orders of magnitude from nominal (20 μ rad on a side). This concept provides a viable comparison to any other tracking algorithm where either the range or the pixel size is different from that which is employed in this algorithm.

The second scenario, scenario #4, consists of the same parameters as the benchmark scenario with the exception that the target pulls a 20-g turn instead of a 10-g turn. This scenario is performed with the intent of establishing the benchmark of performance for a subsequent 20-g simulation and demonstrating the capability of the tracker to maintain good performance against a vehicle pulling a 20-g turn. This performance data proved comparable to previous efforts.

The third and fourth scenarios, scenarios #5 and #7, reflect different realizations of filter-assumed correlation times than had previously been assumed. Both showed that, when tracking a target represented by a correlation time other than that for which the filter was tuned, tracking performance was not severely impacted. However, both cases exhibited a time-increasing bias in the x-channel commencing at the time of the maneuver initiation. Also, the bias tends to worsen as the correlation time gets faster, i.e., represented by a smaller τ . Also recall that the mean squared value of acceleration with respect to the driving

white noise remained equivalent to that in the benchmark case where the slower correlation time had been assumed. This gives rise to the conclusion that, given a bank of filters based upon a particular set of assumed correlation time parameters, if a time-increasing bias in the mean error is prevalent, a possible way to counteract this effect would be to increase the strength of the filter driving white noise, possibly via an adaptive parameter estimation method.

The fifth scenario, scenario #9, demonstrated the relationship between pixel size and filter "Q", given that all other parameters don't change. This demonstration showed that, as the pixel size decreases, the filter white noise strength should decrease by a corresponding amount (both decrease or both increase) if the same basic estimation characteristics are to be maintained.

7.2.4 Target Trajectory Results. The tracking algorithm's performance against two target trajectories not previously considered showed that the algorithm adequately tracked the targets. However, in both cases, the RMS value of the error statistics approached the total of one pixel, the bound upon which good tracking is predicated.

7.3. Investigations of Rotating Rectangular Field-of-View.

Preliminary investigations as to the feasibility of implementing a rotating rectangular field-of-view filter have been conducted. The intent of these investigations was to determine a method by which a rectangular FOV filter

could be rotated so as to align the "elongated" side with the best estimate of the acceleration vector. The reason behind desiring a rotating filter is the possibility of the tracking algorithm to perform its function with four filters rather than five (or even more if one were to allow a larger field-of-view dimension to exist in directions other than pure azimuth and elevation), as had been done previously. The first instinct in pursuing this task was to utilize the estimated acceleration vector, as determined from the multiple model adaptive estimator subroutine embedded in the software, for orienting the longer side of a rectangular field-of-view. Analysis of the data provided from a single Monte Carlo simulation for the Gauss-Markov tracking algorithm against a trajectory #2, 10-g maneuver revealed that utilizing the estimated acceleration was not a good choice. The reason for this is that the estimated acceleration vector is a noisy estimate that can change quite rapidly, and did in fact change sign (direction) from frame to frame at one and one-half seconds into the scenario. This is where the estimated acceleration vector tended to oscillate between the first and second quadrant (of a Cartesian coordinate system) for a period of 17 frames. In addition, from a frame-to-frame evaluation of the respective angular separation of sequential estimated acceleration vectors, rotations of 30 μ rad to 135.31 mrad would be required in order for the FOV to maintain alignment of its "elongated" side with this vector. The inconsistency

associated with estimating the acceleration vector and the corresponding large rotation angle ($135 \text{ mrad} = 0.75 \text{ deg}$) necessary to rotate an array of picture elements compelled further consideration of the rotational concept with respect to following the estimated velocity vector.

Initial evaluation of the estimated velocity vector as a candidate by which to command rotations showed much more stable characteristics than did the estimated acceleration vector. Oscillations from frame to frame were not apparent and angular difference from frame to frame ranged from as small as $10 \text{ } \mu\text{rad}$ to as large as 7.38 mrad . Rather than attempt to align the "elongated" axis with the estimated velocity vector (as was the intent if estimated acceleration were used), aligning the "short" axis of the rectangular FOV with the velocity vector direction is considered since this maintains the acceleration vector along the "elongated" side (assuming that acceleration is predominantly orthogonal to velocity).

Several rotational methods were attempted using different combinations of the filter states, the truth model states, and the hotspot intensity peaks, i.e. rotate filter states, truth model states, and hotspot intensity peaks all in same direction or rotate filter states and truth model states in one direction and rotate hotspot intensity peaks in the opposite direction. All evaluations of these rotations provided similar results. The tracking algorithm

performed well initially and then tended to "lose track" for the majority of the simulations. This problem stemmed from the fact that, although the states from the current frame are rotated, the rotated image array generated from the current frame is correlated with the "smoothed" image (template) from the previous frame which is not rotated with respect to the current frame. This correlation of the present data with the previous image (smoothed) is the mechanism by which the filter measurements are extracted. In effect, what has been shown is that, if the currently generated rotated image is correlated with an unrotated image (or otherwise interpreted as delaying the rotation by a sample period), the performance capability of the Gauss-Markov tracking algorithm degrades significantly. This is, in effect, a suboptimal implementation of the rotating FOV method. One means of compensating for the time lag would be to rotate the template (previously generated image) by the same amount, and at the same time, as the filter estimates. However, if the FLIR image plane is physically rotated by an angle θ , the image in that plane appears to rotate by $-\theta$; therefore, the simulation of the true image rotation by θ corresponds to the tracker's x and y axes rotated with respect to the original orientation and the reconstructed image requiring a rotation by $-\theta$.

A method of implementing a rotation to a spatial image array can be found in (6). This method transforms an $N \times N$ x-y spatial array to a $2N \times N/2$ R-theta plane array as

depicted by Figure 7.1. Each row in the R-Theta plane corresponds to an angular displacement in the x-y plane from the center of the plane; each row (or each elemental increment in the horizontal direction) equates to an increment of π/N radians if an image is considered to cover

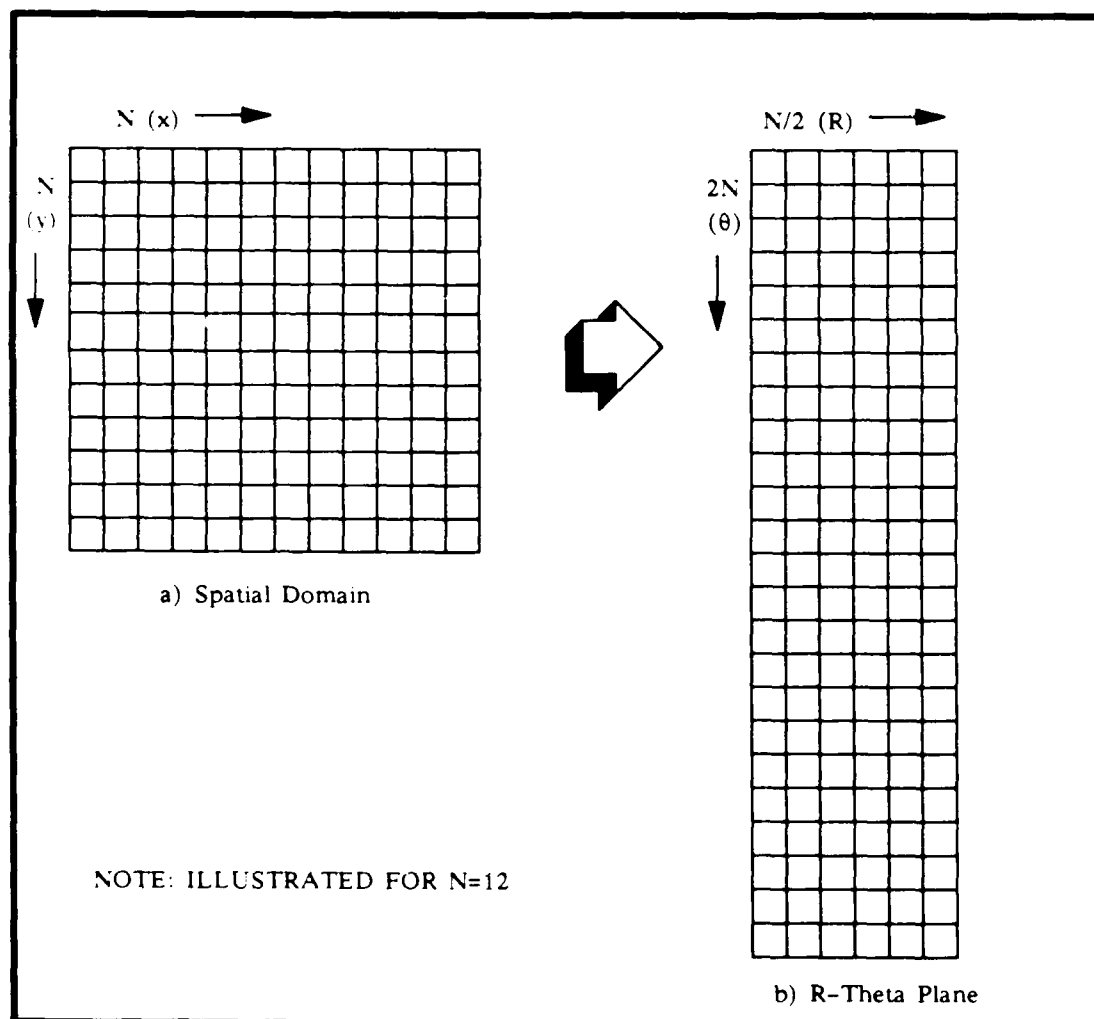


Figure 7.1 Spatial Domain to R-Theta Transformation

360° in the x-y plane (symmetry can be exploited if the x-y plane lies in the Fourier domain, which is presently not considered, and, in this case, an image can be considered to cover 180° in the mapped space (x-y plane) and accompanied by its complex conjugate).

Rotations are simply performed in the R-Theta plane by shifting rows up or down by the number of incremental angles that correspond to the desired angle of rotation. One would expect not to be able to rotate by exactly a desired amount, but to the "closest" incremental rotation possible based upon the number of "N" discrete quantities. Since the image array in the GM tracking algorithm is represented by a 24 X 24 array, each row, if this method were to be employed, would correspond to an increment of $\pi/48$ radians. To implement such a rotational algorithm for this tracker, preliminary investigations have disclosed that angle increments in Theta on the order of 1 to 3 mrad would be necessary for the filters to be able to maintain "lock" on a frame-to-frame basis. One topic discussed in the following section is a proposed implementation technique that may warrant further investigation.

7.4 Recommendations for Further Study

The following recommendations are suggested for further study in enhancing the FLIR tracking system:

1. Further investigation of the effects of the bending/vibration phenomenon may be necessary. If the level

of bending/vibration is actually greater than that which is expected (see Appendix A), additional filter tuning should be performed (taking into consideration the bias effect) or the possibility of modeling the bending effects in the filter model should be conducted. Representing the bending effects in the filter model via reduced order modeling should be considered.

2. Implementing the rotating rectangular field-of-view should be investigated. One possible means of rotating the image template would be to reconstruct the image array consisting of a 24 X 24 intensity data array into a 96 X 96 intensity data array (effectively transforming each element in the 24 X 24 array into 4 elements in the 96 X 96 array) and perform the R-Theta transformation previously described. This redefined image could then be rotated in increments of $\pi/192$, or 3.27 mrad. Although the 96 X 96 array definition provides a good starting point for possible implementation, it is certainly not the only possible choice. The image could then be inverse transformed back to the x-y spatial domain and redefined to a 24 X 24 intensity data array to be correlated with the currently constructed image.

3. The problem of initial target acquisition should be investigated. Since accurate velocity and acceleration estimates are not available a priori (as was assumed in this effort), they should be determined on-line. One potential solution would be to estimate velocity and acceleration by

the method: $v_x = \Delta x / \Delta t$ and $a_x = \Delta v_x / \Delta t$, where the x-centroid and y-centroid locations at adjacent sample times could be used to determine Δx and Δy . Also, Δt should be made as small as computation time will allow and still minimize the time-to-acquisition. Another potential solution would be to employ the inverse covariance Kalman filter propagation and update forms described in Maybeck (9) to estimate initial position, velocity, and acceleration variables on the basis of a batch of N measurements (no a priori information assumed, i.e., $P_0^{-1} = 0$), and then switch back to the conventional form once P^{-1} becomes nonsingular.

4. Still requiring investigation for this tracking algorithm is an indepth target/decoy sensitivity analysis. Sensitivity to the following parameters should be pursued: (1) decoy intensity, shape, and size, (2) separation from the target during decoy ignition, and (3) tracking time prior to decoy release. Image and/or filter residual characteristics should be analyzed in order to suggest ways in which decoys may be distinguished from targets.

APPENDIX A

DERIVATION OF THE BENDING AND VIBRATION
EFFECTS OF A LARGE SPACE STRUCTURE

Appendix A

Mechanical Truth States Development

The mechanical bending/vibration model used in this study was based on a report prepared for AFWL by the staff of R & D Associates and of Cambridge Research (15). This model is by no means the only possible structural configuration for a space-based structure; however, it does provide the necessary mechanical considerations felt necessary to describe the effects of bending and vibration at several locations about a typical large structure. The predominant sources of random excitation in a space based structure are caused by (15:4-1):

1. Coolant flow through the mirrors, the resonator power management equipment and the associated supply systems;
2. Fluctuating combustion chamber pressures;
3. Fluctuating pressure generated by the laser exhaust flows;
4. Conditioning and transfer of laser reactants; and
5. Steady state operation of the control moment gyros.

The power spectral density (PSD) curve of Figure 2.2, which is shown on the following page as Figure A.1, was determined to be the predominant contributor to the bending/vibration effects on a tracker's line of sight, with the response in both the x- and y- directions being similar. Although this PSD response represents the disturbance seen at the secondary mirror LOS, a similar response with approximately

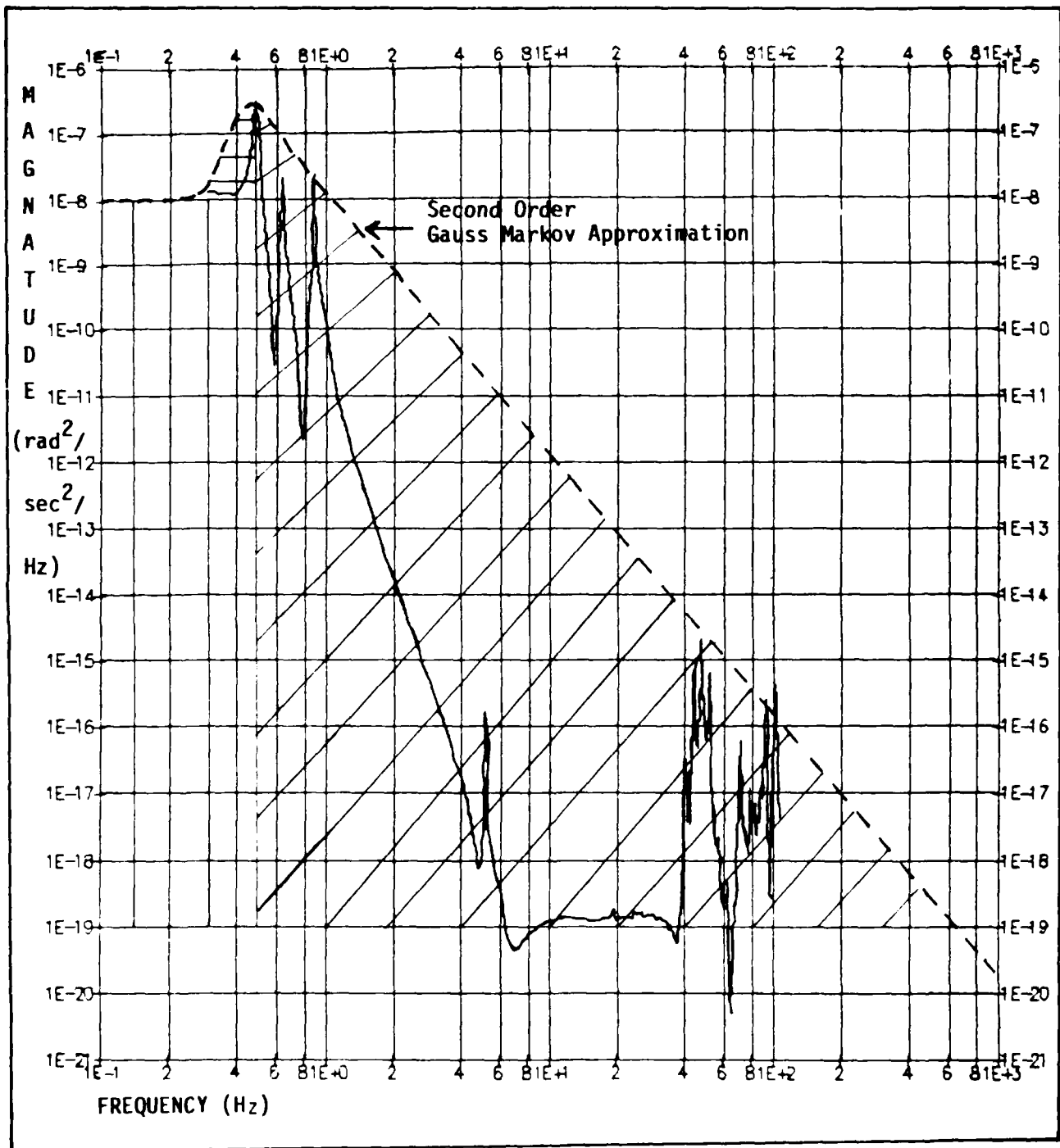


Figure A.1. Dominant Vibration Response Spectrum.

a 60 dB isolation between the two is expected at the tracker's LOS (5). This research effort approximates the response of the system due to the PSD curve of Figure A.1 to be represented by a second order Markov process described by the transfer function:

$$G(s) = \frac{x_b}{w_b} = \frac{K_b \omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2} \quad (A-1)$$

where:

ω_n = bending/vibration mode natural frequency, = .5 Hz.
= 3.14159264 rad./sec.

K_b = gain adjusted to obtain the desired root mean square (RMS) bending/vibration output, σ_{bv} :
where σ_{bv} is defined later in Equation (A-23)

ζ = damping ratio, = .15 (17:359)

The roots of the characteristic equation can be determined by solving the quadratic equation formed by setting the denominator of Equation (A-1) to zero, resulting in:

$$s_{1,2} = \sigma_b + j\omega_b = -.47124 \pm j3.10605 \quad (A-2)$$

The corresponding state space representation in standard observable form for both directions would appear as:

$$\begin{bmatrix} \dot{x}_{b1} \\ \dot{x}_{b2} \\ \dot{y}_{b1} \\ \dot{y}_{b2} \end{bmatrix} = \begin{bmatrix} F_{11} & F_{12} & 0 & 0 \\ F_{21} & F_{22} & 0 & 0 \\ 0 & 0 & F_{11} & F_{12} \\ 0 & 0 & F_{21} & F_{22} \end{bmatrix} \begin{bmatrix} x_{b1} \\ x_{b2} \\ y_{b1} \\ y_{b2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ G_b & 0 \\ 0 & 0 \\ 0 & G_b \end{bmatrix} \begin{bmatrix} w_{bx} \\ w_{by} \end{bmatrix} \quad (A-3)$$

and:

$$\underline{z}_b = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_{b1} \\ x_{b2} \\ y_{b1} \\ y_{b2} \end{bmatrix} \quad (\text{A-4})$$

where:

$$F_{11} = 0$$

$$F_{12} = 1$$

$$F_{21} = -\omega_n^2 = -9.647547 \text{ rad.}^2/\text{sec.}^2$$

$$F_{22} = -2\zeta\omega_n = -.94248 \text{ rad./sec.}$$

$$G_b = K_b\omega_n^2 = 9.647547K_b$$

Equations (A-3) and (A-4) are of the form:

$$\dot{\underline{x}} = \underline{F}\underline{x} + \underline{G}\underline{w} \quad (\text{A-5})$$

and

$$\underline{z} = \underline{H}\underline{x} \quad (\text{A-6})$$

The state transition matrix, Φ_b , can be derived by employing the following relationship:

$$\Phi_b(\Delta t) = L^{-1}\{(sI - \underline{F}_b)^{-1}\} \Big|_{\Delta t} \quad (\text{A-7})$$

where:

L = the Laplace operator

I = the identity matrix

Δt = the sample period, $t_{i+1} - t_i$

Performing the operation defined by Equation (A-7) on \underline{F}_b results in the following form for the bending phenomena state transition matrix:

$$\Phi_b(\Delta t) = \begin{bmatrix} \phi_{b1} & \phi_{b2} & 0 & 0 \\ \phi_{b3} & \phi_{b4} & 0 & 0 \\ 0 & 0 & \phi_{b1} & \phi_{b2} \\ 0 & 0 & \phi_{b3} & \phi_{b4} \end{bmatrix} \quad (A-8)$$

where:

$$\begin{aligned} \phi_{b1} &= \exp(-\sigma_b \Delta t) [\cos(\omega_b \Delta t) + (\sigma_b / \omega_b) \sin(\omega_b \Delta t)] \\ &= .9945787985 \end{aligned}$$

$$\begin{aligned} \phi_{b2} &= \exp(-\sigma_b \Delta t) [(1/\omega_b) \sin(\omega_b \Delta t)] \\ &= .03275523095 \end{aligned}$$

$$\begin{aligned} \phi_{b3} &= \exp(-\sigma_b \Delta t) \{-[1 + (\sigma_b / \omega_b)^2] \sin(\omega_b \Delta t)\} \\ &= -.1040812213 \end{aligned}$$

$$\begin{aligned} \phi_{b4} &= \exp(-\sigma_b \Delta t) [\cos(\omega_b \Delta t) - (\sigma_b / \omega_b) \sin(\omega_b \Delta t)] \\ &= .9637076486 \end{aligned}$$

A discrete-time model of the propagation of \underline{x}_b is given by:

$$\underline{x}_b(t_{i+1}) = \Phi_b \underline{x}_b(t_i) + \underline{w}_{bd} \quad (A-9)$$

where:

$$\underline{w}_{bd} = \int_{t_i}^{t_{i+1}} \Phi_b(t_{i+1}, \tau) \underline{G}_b(\tau) \underline{w}_b(\tau) d\tau \quad (A-10)$$

and \underline{w}_{bd} has the following statistics:

$$E[\underline{w}_{bd}(t_i)] = 0 \quad (A-11)$$

$$E[\underline{w}_{bd}(t_i) \underline{w}_{bd}^T(t_j)] = \underline{Q}_{bd} \delta_{ij} \quad (A-12)$$

and:

$$\underline{Q}_{bd} = \int_{t_i}^{t_{i+1}} \Phi_b(t_{i+1}, \tau) \underline{G}_b(\tau) \underline{Q}_b(\tau) \underline{G}_b^T(\tau) \Phi_b^T(t_{i+1}, \tau) d\tau \quad (A-13)$$

$$= \begin{bmatrix} Q_{bd1} & Q_{bd2} & 0 & 0 \\ Q_{bd3} & Q_{bd4} & 0 & 0 \\ 0 & 0 & Q_{bd1} & Q_{bd2} \\ 0 & 0 & Q_{bd3} & Q_{bd4} \end{bmatrix} \quad (A-14)$$

where:

$$Q_{bd1} = \int_{t_i}^{t_{i+1}} G_b^2 \phi_{b2}^2 d\tau = .3254294713 K_b^2$$

$$Q_{bd2} = \int_{t_i}^{t_{i+1}} G_b^2 \phi_{b2} \phi_{b4} d\tau = Q_{bd3} = .01038495095 K_b^2$$

$$Q_{bd4} = \int_{t_i}^{t_{i+1}} G_b^2 \phi_{b4}^2 d\tau = .3055407229 K_b^2$$

Note that the elements of Q_{bd} have been expressed in terms of the power spectral density gain parameter, K_b . Although current technology assessments on large space structure mechanical disturbances indicate radial displacement levels consistent with those of Figure A.1, this parameter can easily be varied to analyze the level of disturbance that would degrade filter performance.

For the simulation implementation, the discrete-time propagation equation was implemented in the following form:

$$\underline{x}_b(t_{i+1}) = \underline{\phi}_b \underline{x}_b(t_i) + \sqrt{Q_{bd}} \underline{w}_n(t_i) \quad (A-15)$$

where:

$\sqrt{Q_{bd}}$ = Cholesky square root matrix of Q_{bd}

$w_n(t_i)$ = vector of independent, unit-variance, gaussian noises

In order to determine the value of K_b , the form of the covariance matrix must be known. If uncorrelatedness between the x- and y- directions are assumed, then the covariance matrix appears as:

$$P_b = \begin{bmatrix} P_{b1} & P_{b2} & 0 & 0 \\ P_{b3} & P_{b4} & 0 & 0 \\ 0 & 0 & P_{b1} & P_{b2} \\ 0 & 0 & P_{b3} & P_{b4} \end{bmatrix} \quad (A-16)$$

Now, if the time derivative of the covariance matrix is set equal to zero, the steady state value of K_b can be derived by solving the following relationship:

$$\dot{P}_b = F_b P_b + P_b F_b^T + G_b Q_b G_b^T = 0 \quad (A-17)$$

which yields a 4 X 4 block diagonal matrix where each 2 X 2 block is of the form:

$$0 = \begin{bmatrix} 2P_{b2} & P_{b4} + F_{b3}P_{b1} + F_{b4}P_{b2} \\ P_{b4} + F_{b3}P_{b1} + F_{b4}P_{b3} & 2F_{b3}P_{b2} + 2F_{b4}P_{b4} + G_b^2 \end{bmatrix} \quad (A-18)$$

This implies that:

$$P_{b2} = 0 = P_{b3} \quad (A-19)$$

Two equations with three unknown values (P_{b1} , P_{b2} , and G_b) remain:

$$P_{b4} + F_{b3}P_{b1} = 0 \quad (A-20)$$

$$2F_{b4}P_{b4} + G_b^2 = 0 \quad (A-21)$$

Another equation relating one or more of the unknowns, G_b , P_{b1} , or P_{b4} , is necessary in order to accommodate a solution to the above relationship. The variance of the output provides such a relationship:

$$\begin{aligned} P_{zz} &= E\{zz^T\} \\ &= E\{Hx_b x_b^T H^T\} \\ &= \begin{bmatrix} E\{x_{b1}^2\} & 0 \\ 0 & E\{y_{b1}^2\} \end{bmatrix} \end{aligned} \quad (A-22)$$

Making use of the fact that:

$$E\{x_{b1}^2\} = E\{y_{b1}^2\} = P_{b1} = \sigma_{bv}^2 \quad (A-23)$$

and σ_{bv}^2 is equal to the area under the PSD curve which is approximated by the following equation (a geometrical approximation to the area of Figure A.1):

$$\begin{aligned} \sigma_{bv}^2 &= (.5)(K_b^2) + (1/2)(.2)(14K_b^2) \\ &\quad + (1/2)(700)(15K_b^2) \\ &= 5251.9(K_b^2) \text{ rad.}^2/\text{sec.}^2 \end{aligned} \quad (A-24)$$

where:

$$K_b^2 = \text{PSD magnitude at zero frequency}$$

The first term in this approximation is represented by the area in Figure A.1 shaded by horizontal lines, the second term by the area shaded by vertical lines, and the third term by the diagonal lines (Note: levels below 1E-19 are ignored). Equation (A-24) is used to solve Equation (A-20) for P_{b4} in terms of the zero frequency PSD value, K_b :

$$P_{b4} = \omega_n^2 \sigma_{bv}^2 = 5.183E+04(K_b^2) \text{ rad.}^4/\text{sec.}^4 \quad (\text{A-25})$$

The derivations thus far have defined all of the parameters necessary to perform a time propagation of the states of the bending/vibration model. In order to provide measurement update relationships compatible with the tracking algorithm, the measurement variables should be expressed in terms of pixels. Since the propagation variables are expressed in terms of radians, a simple conversion from radians to pixels is necessitated, with the resulting output relationship appearing as:

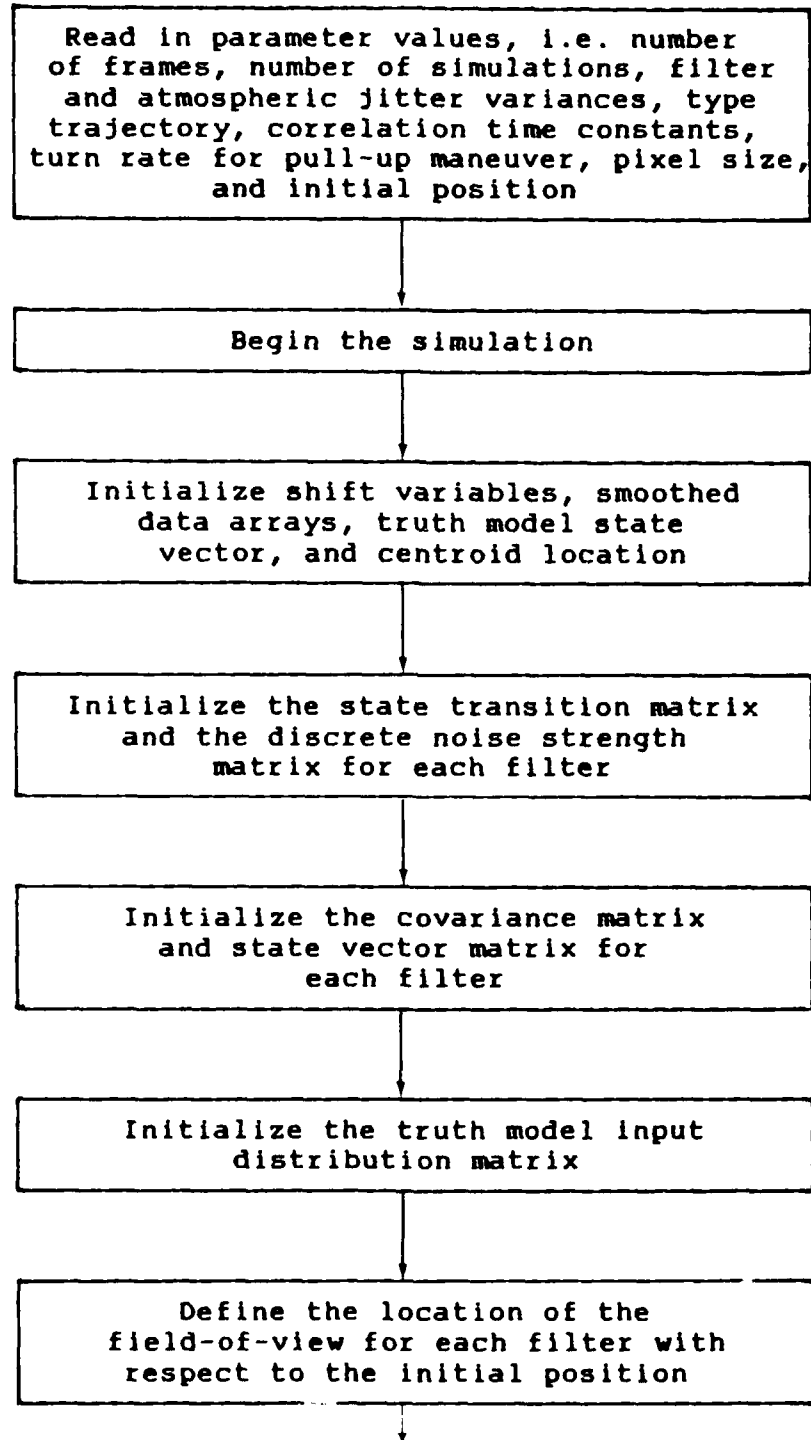
$$y_b = \begin{bmatrix} k_p & 0 & 0 & 0 \\ 0 & 0 & k_p & 0 \end{bmatrix} x_b \quad (\text{A-26})$$

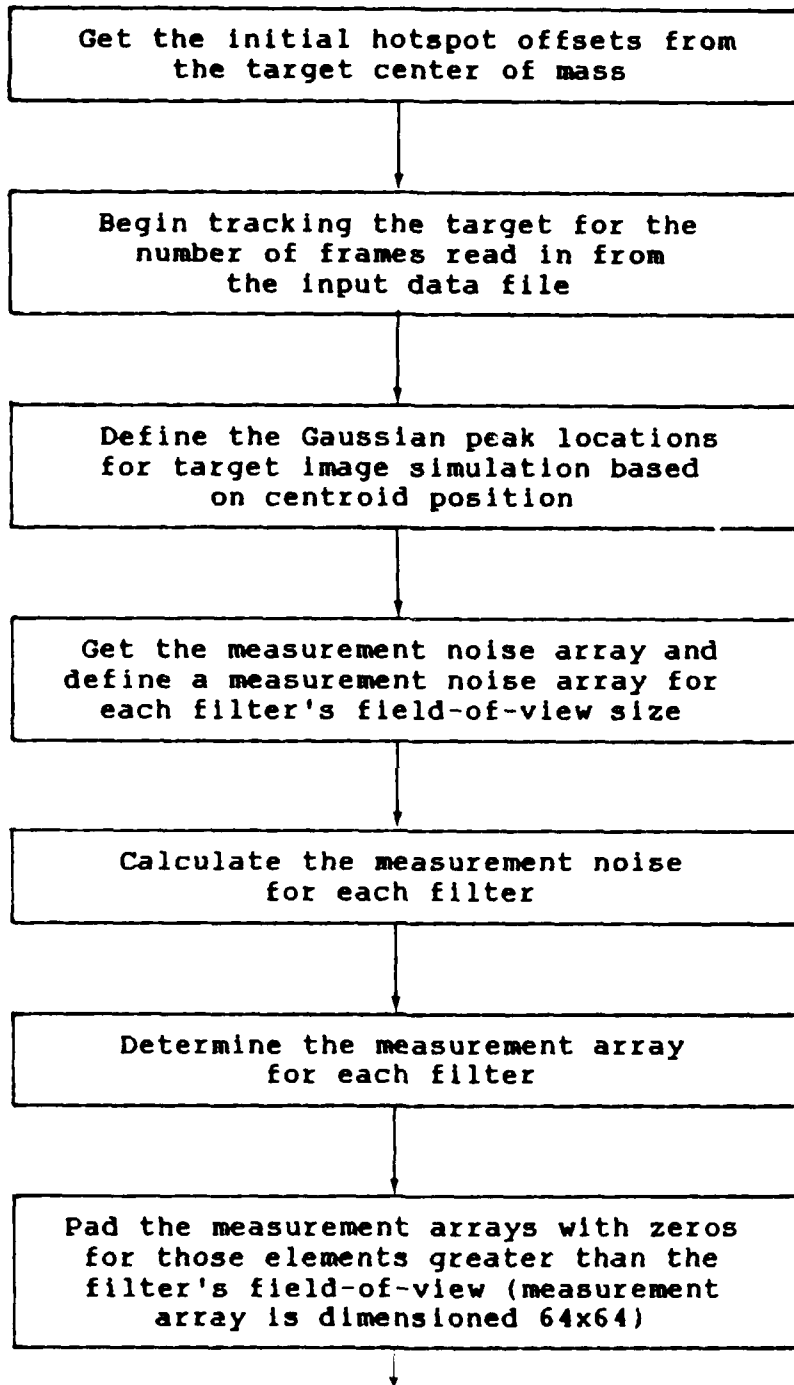
APPENDIX B

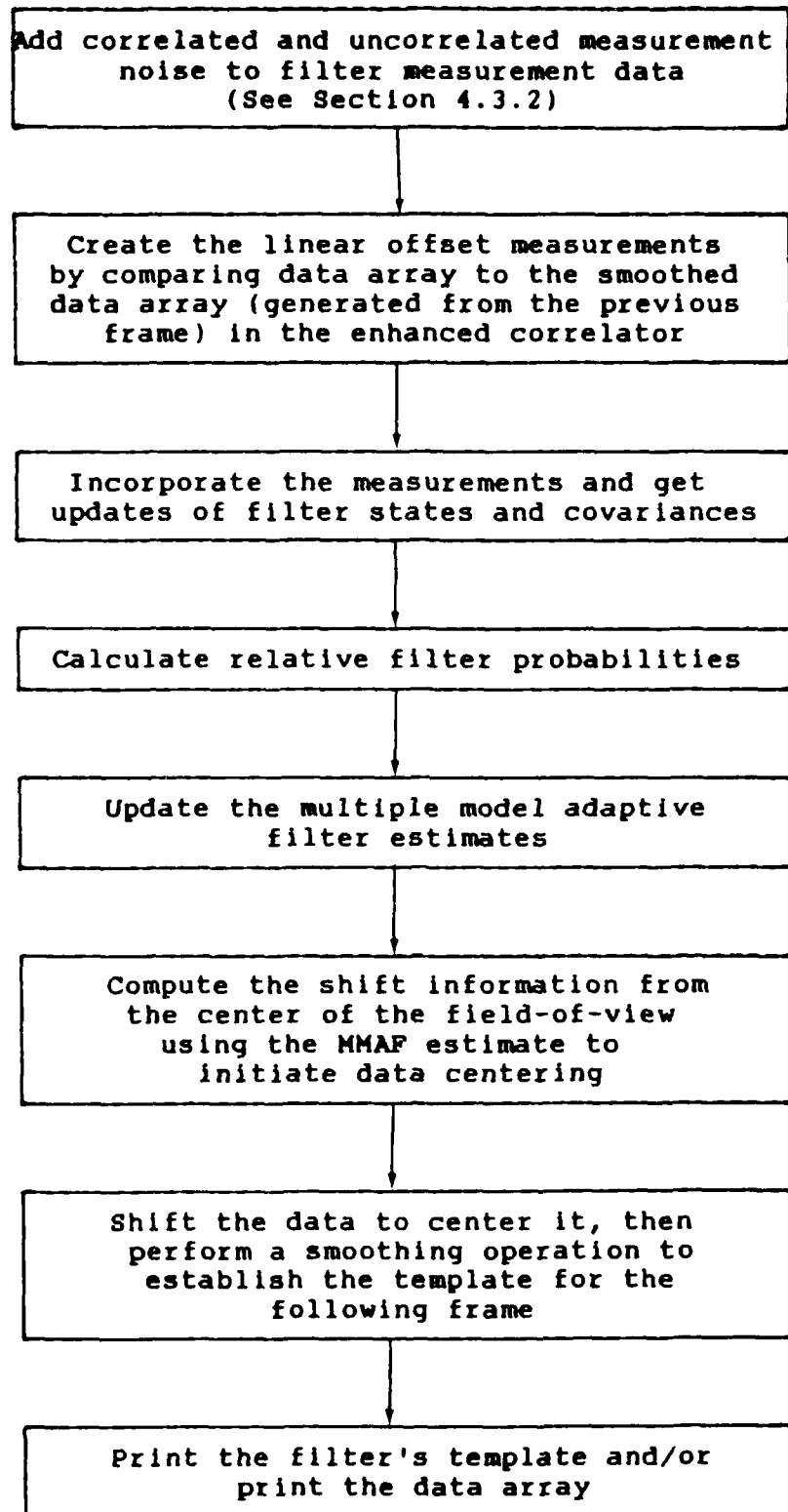
SIMULATION PROGRAM FLOW DIAGRAM

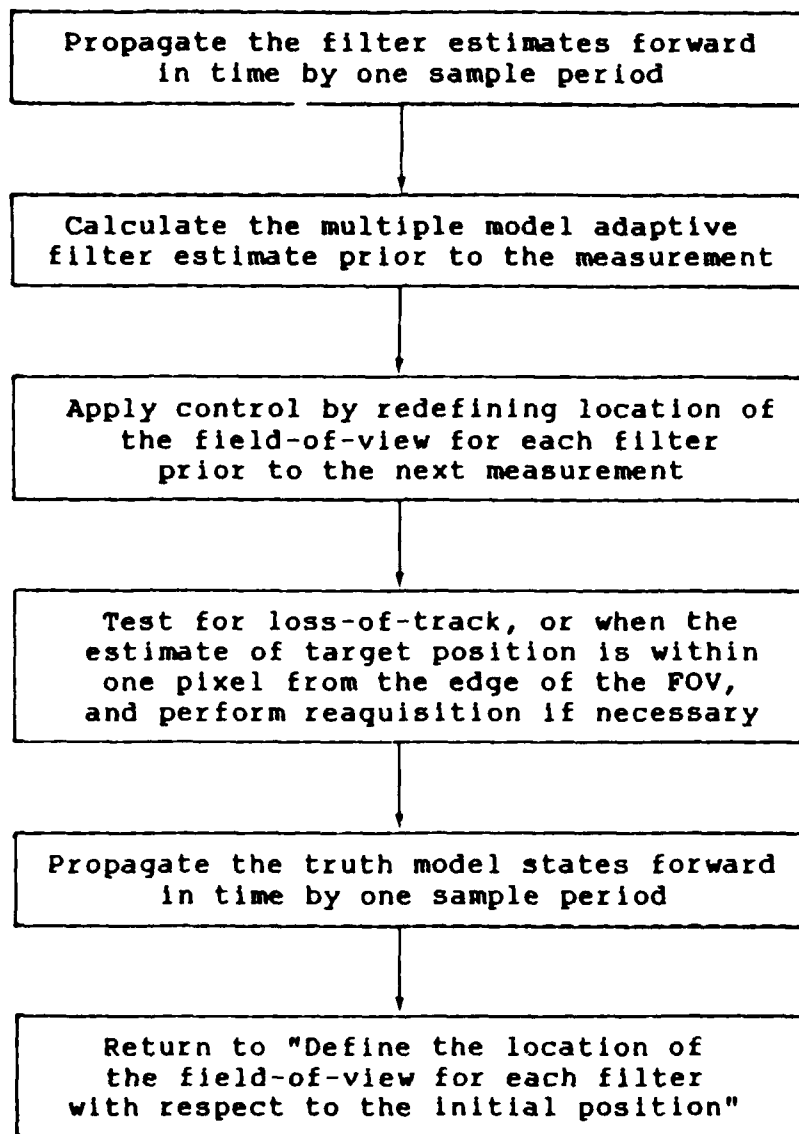
Appendix B

Computer Simulation Flow Diagram









APPENDIX C

SINGLE FILTER SIMULATIONS
BENCHMARK AT 30 Hz AND 50 Hz
(REFERENCED FROM SECTION 6.2)

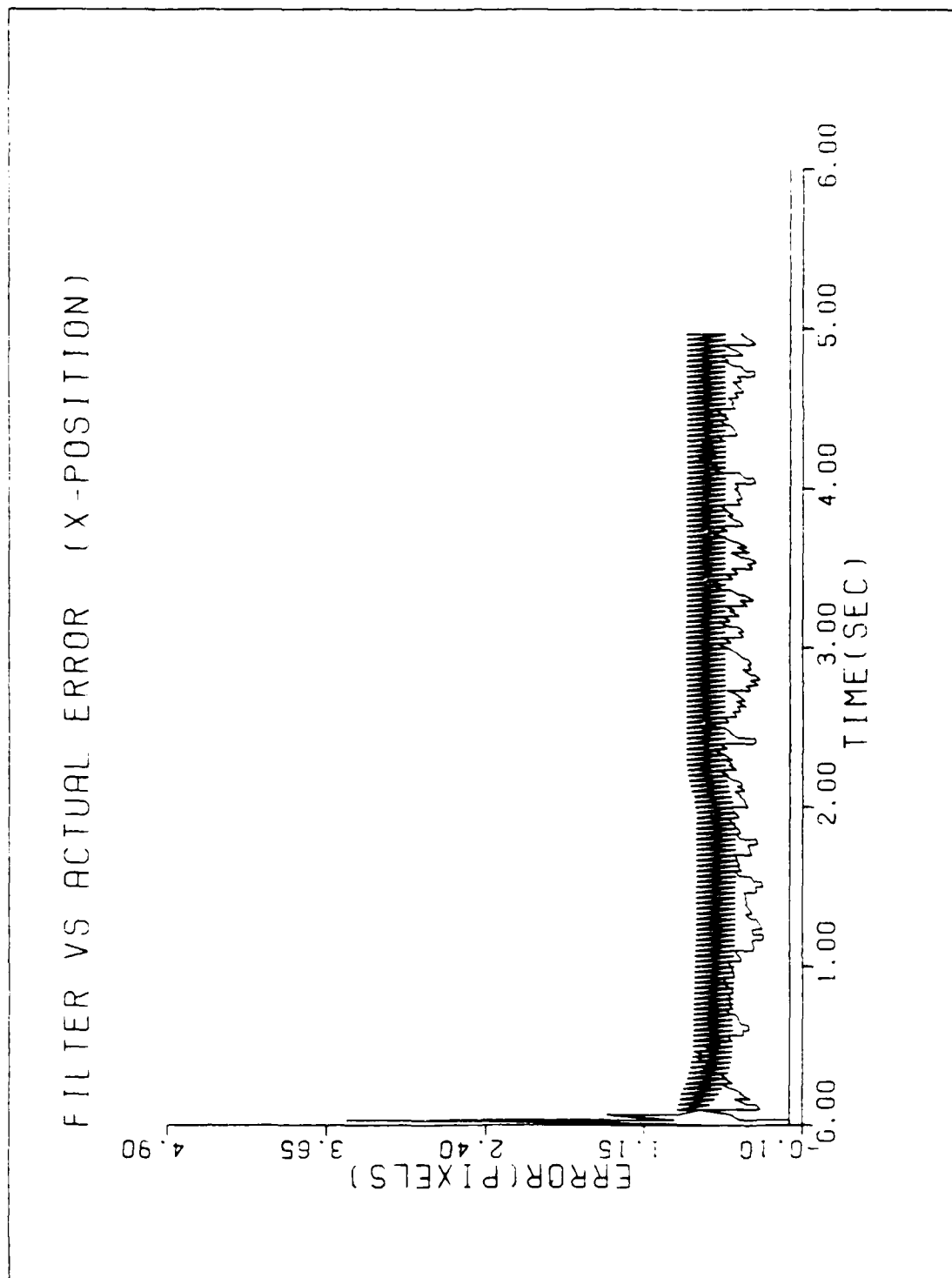


Figure C-1. GM/SNG-S/T2/10-G

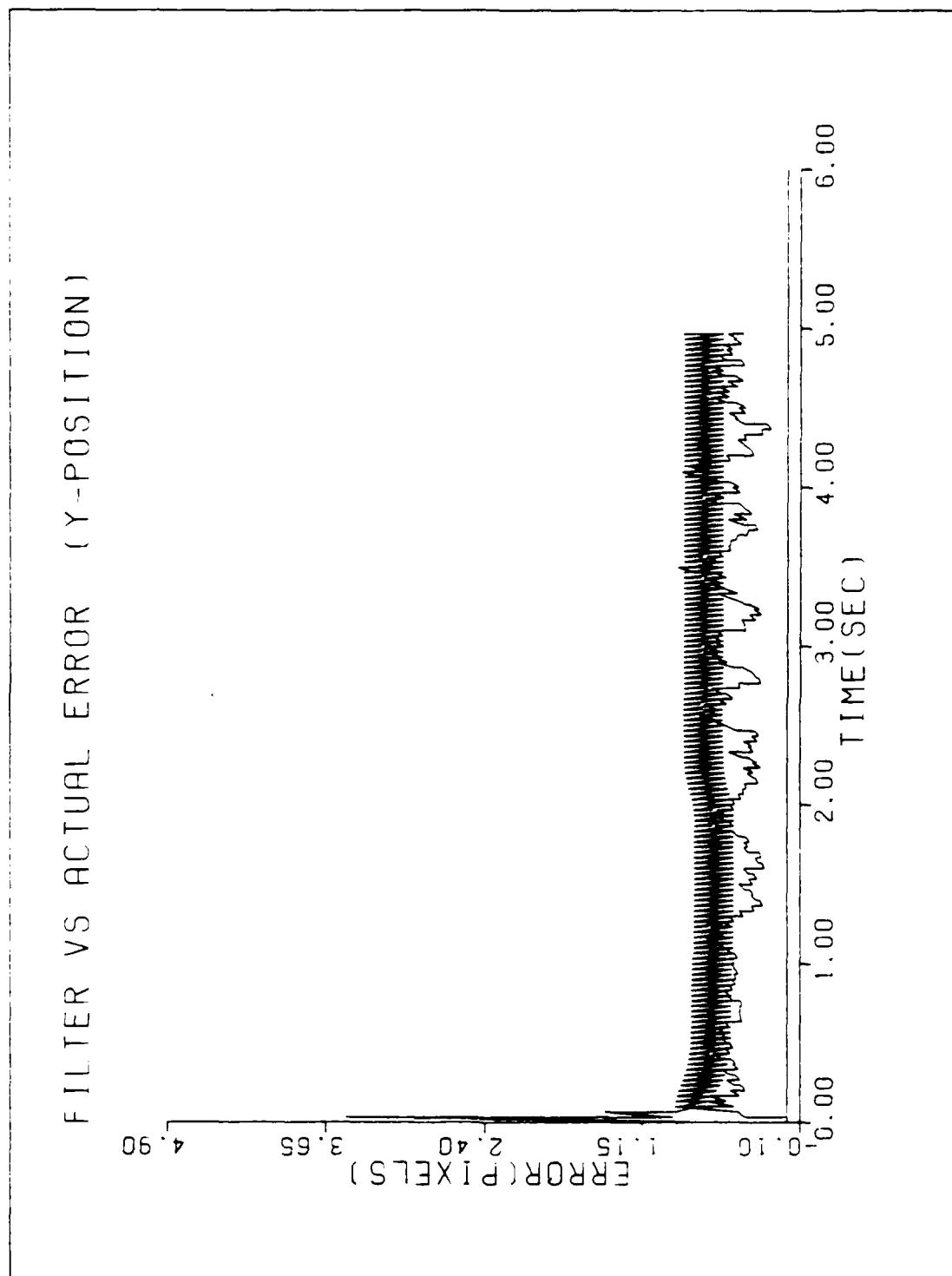


Figure C-2. GM/SNG-S/T2/10-G

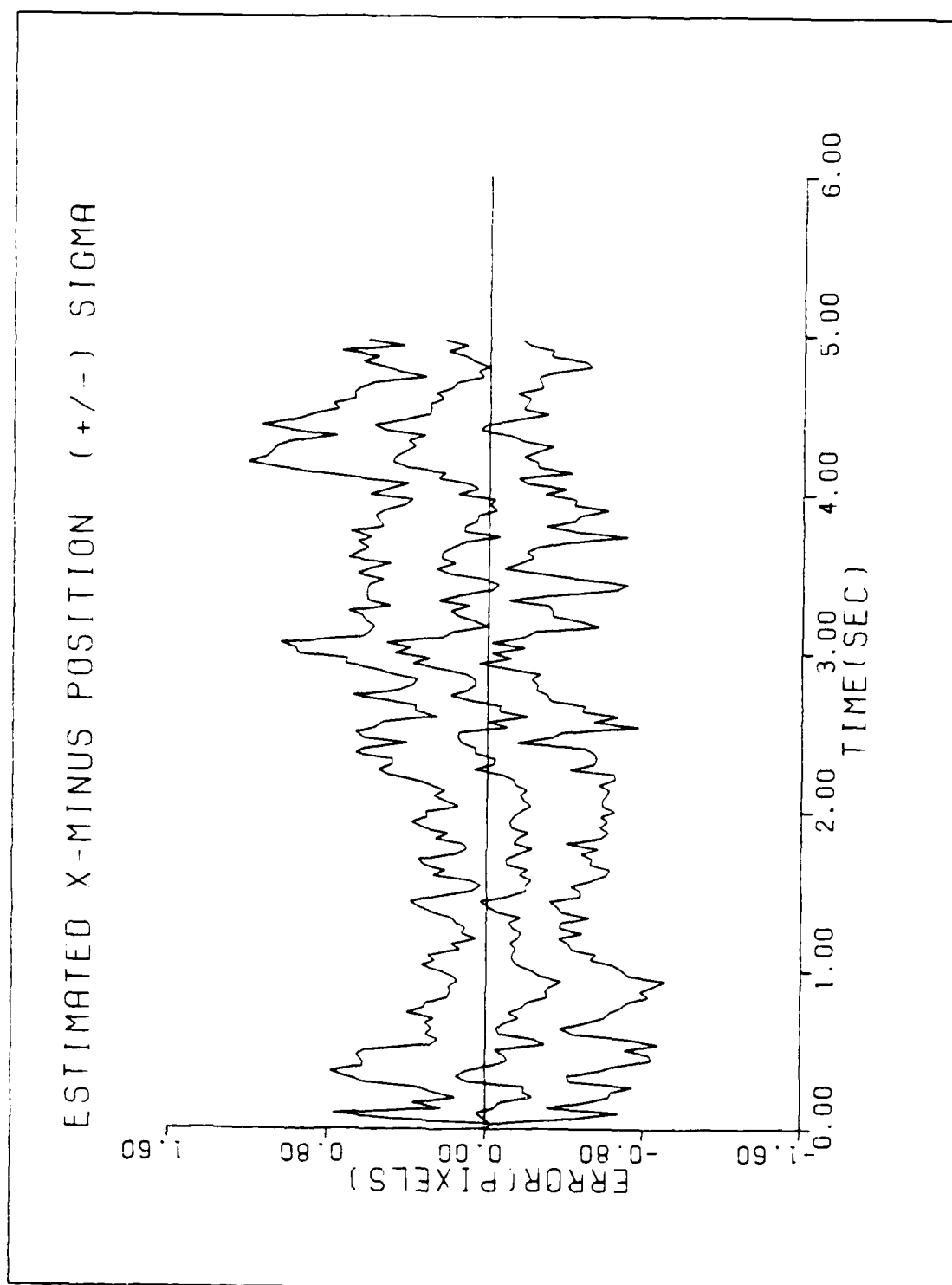


Figure C-3. GM/SNG-S/T2/10-G

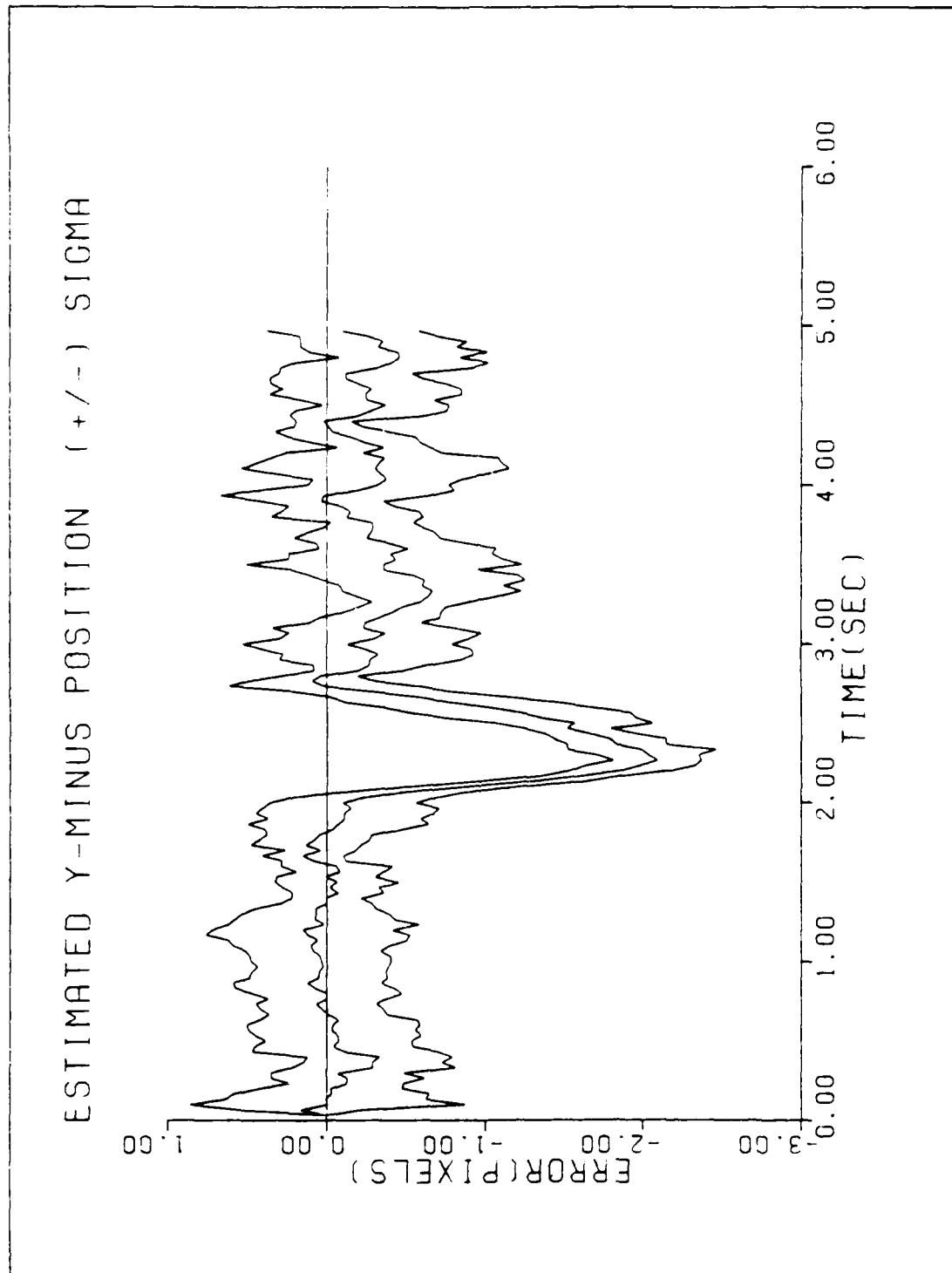


Figure C-4. GM/SNG-S/T2/10-G

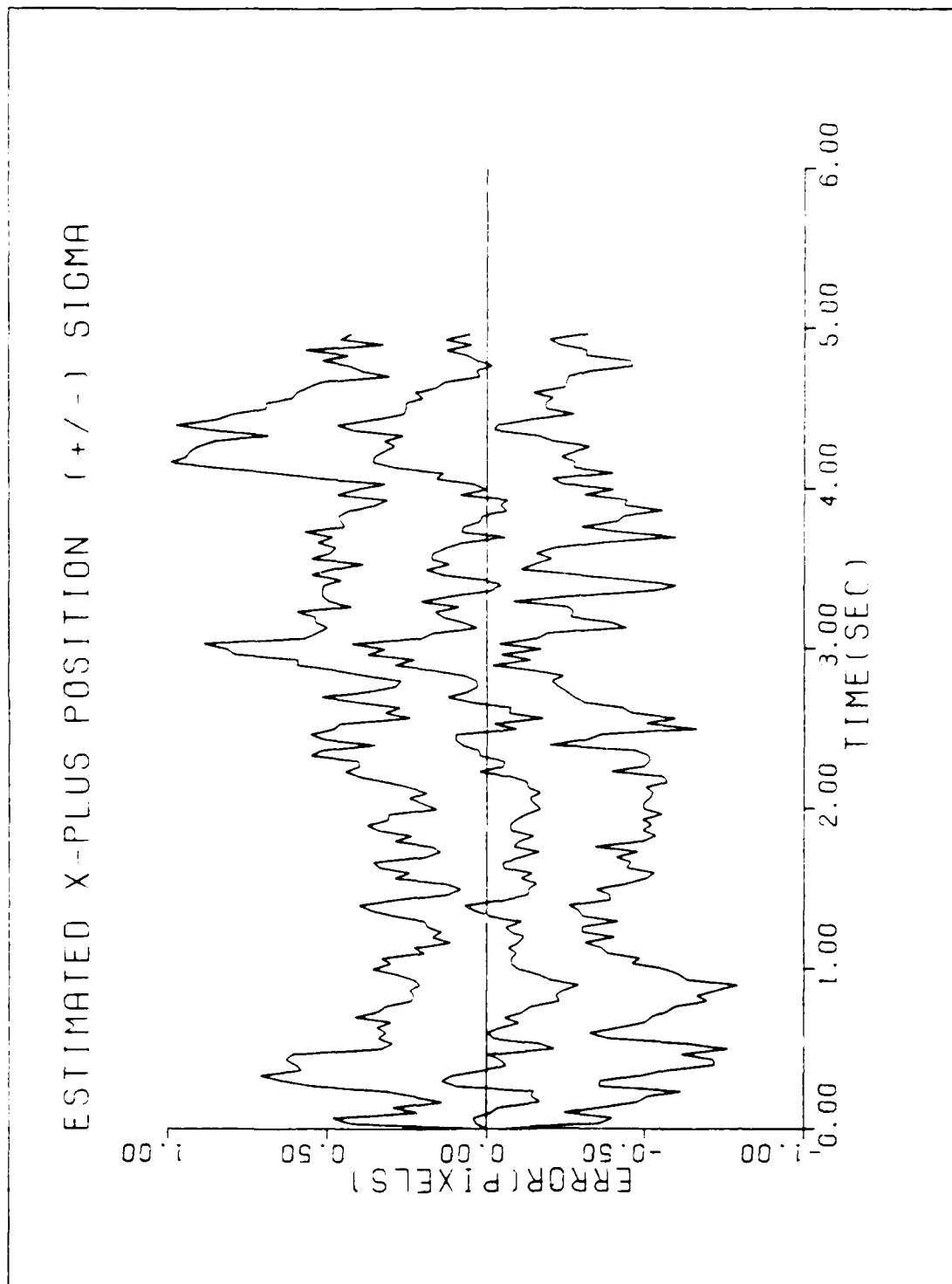


Figure C-5. GM/SNG-S/T2/10-G

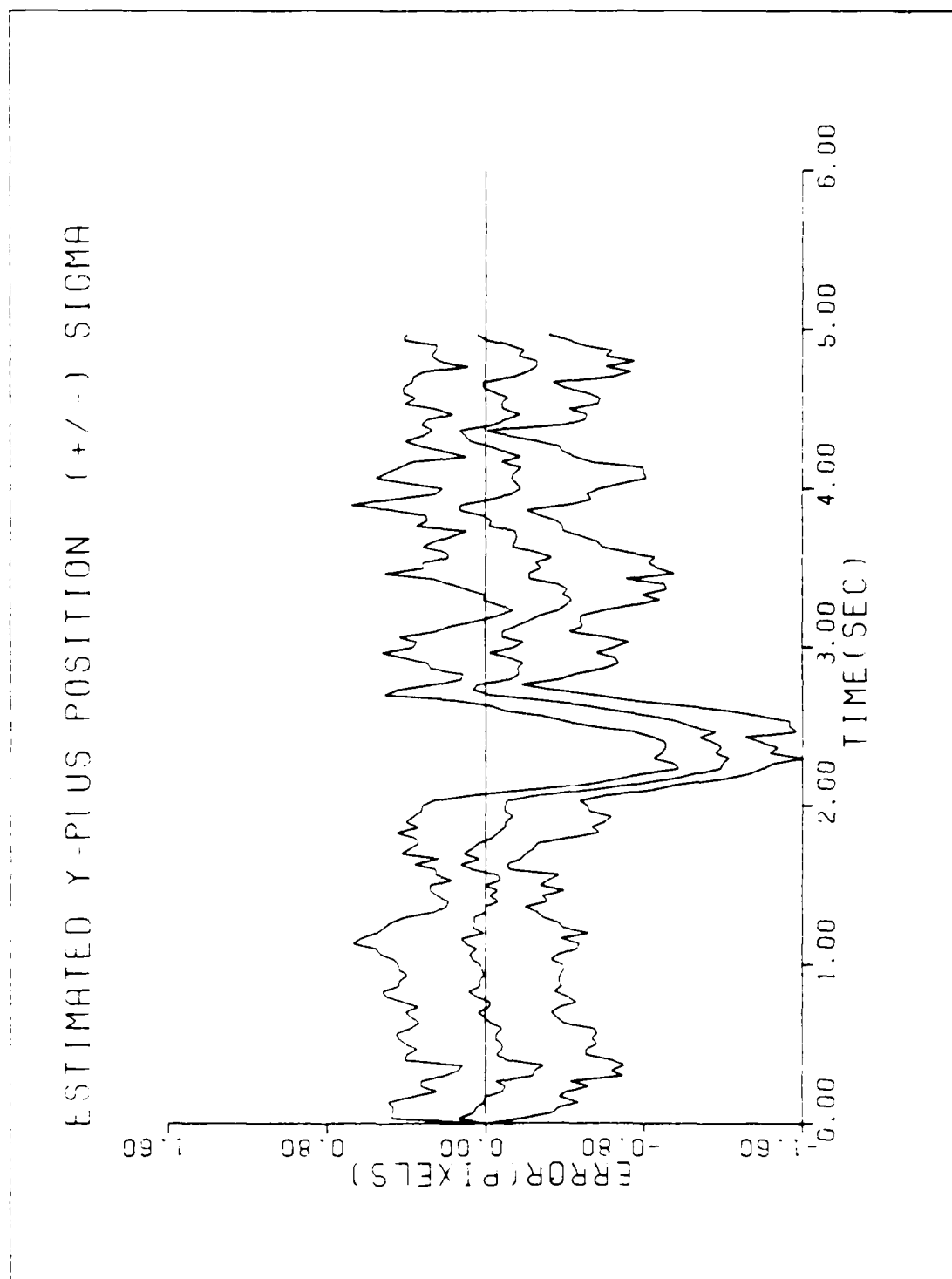


Figure C-6. GM/SNG-S/T2/10-G

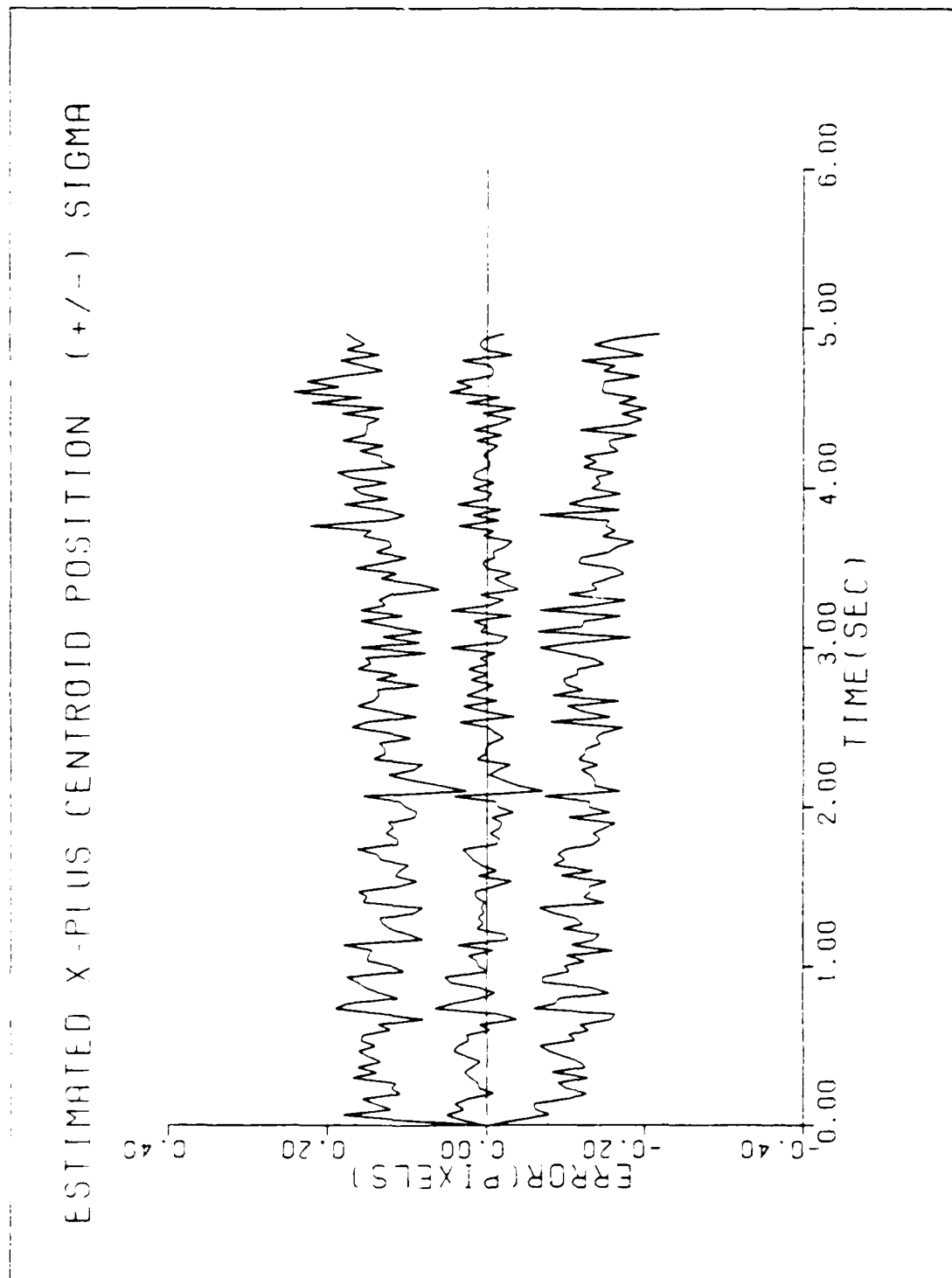


Figure C-7. GM/SNG-S/T2/10-G

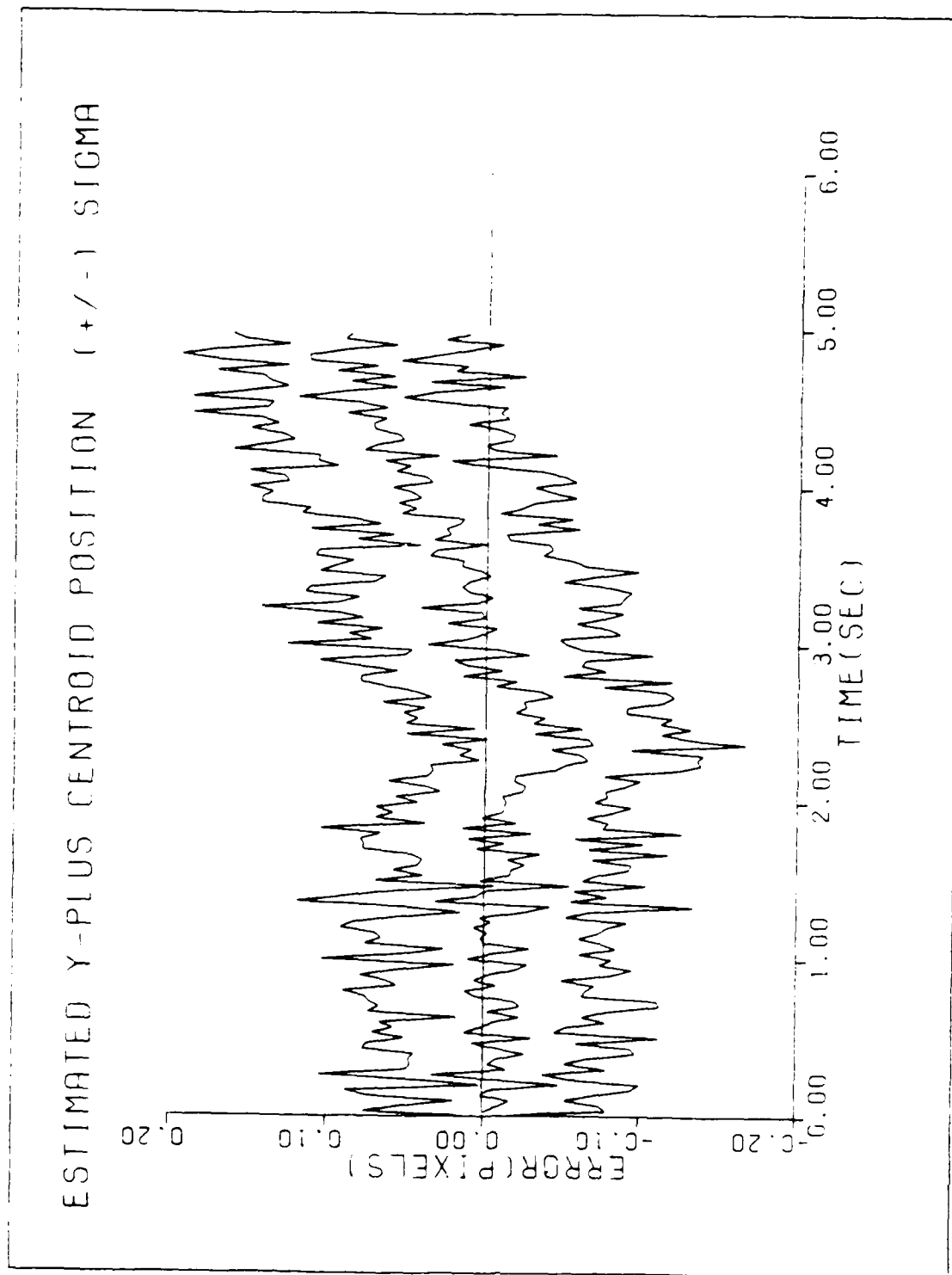


Figure C-8. GM/SNG-S/T2/10-G

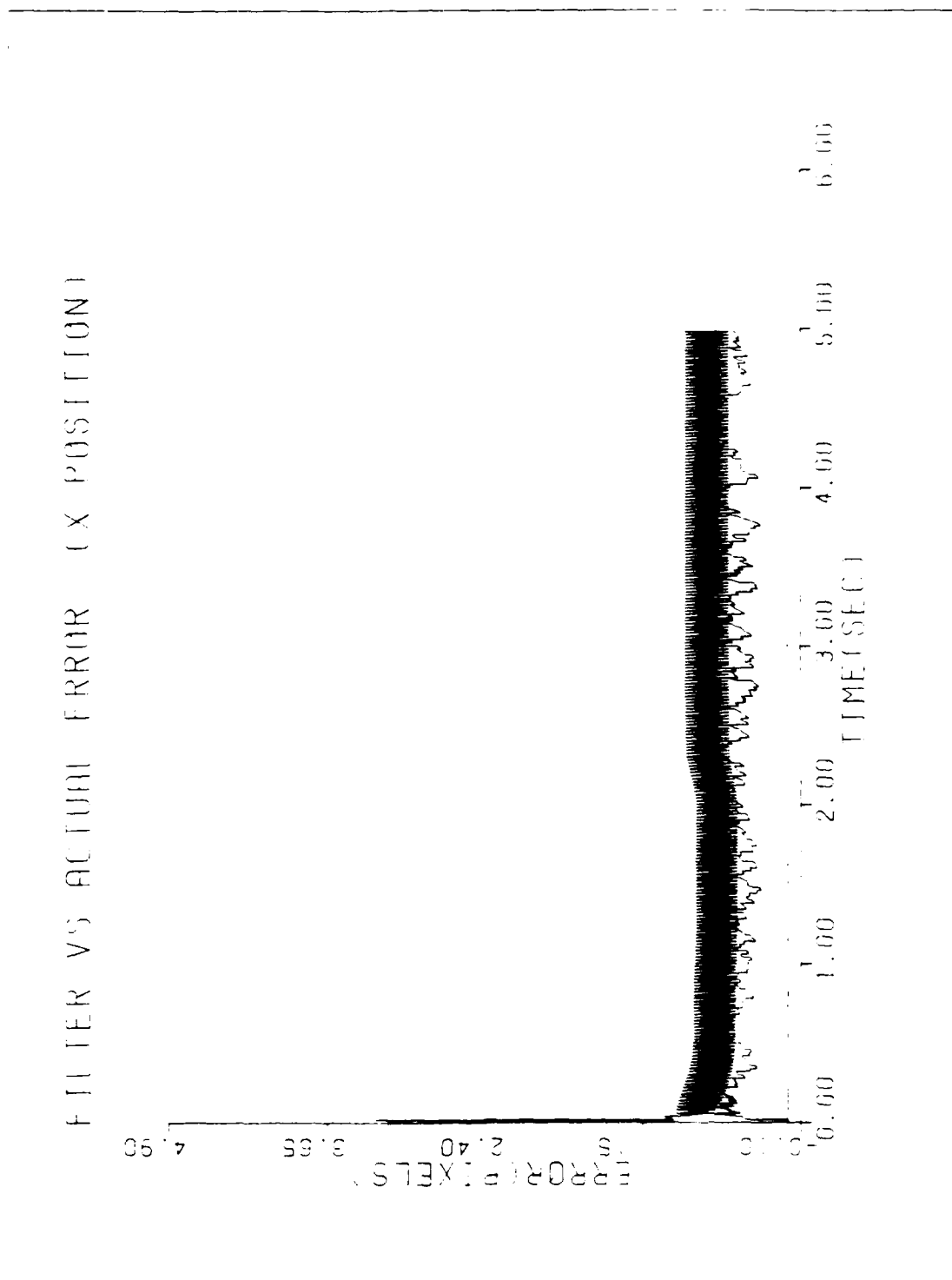


Figure C-9. GM/SNG-S/T2/10-G/50 Hz

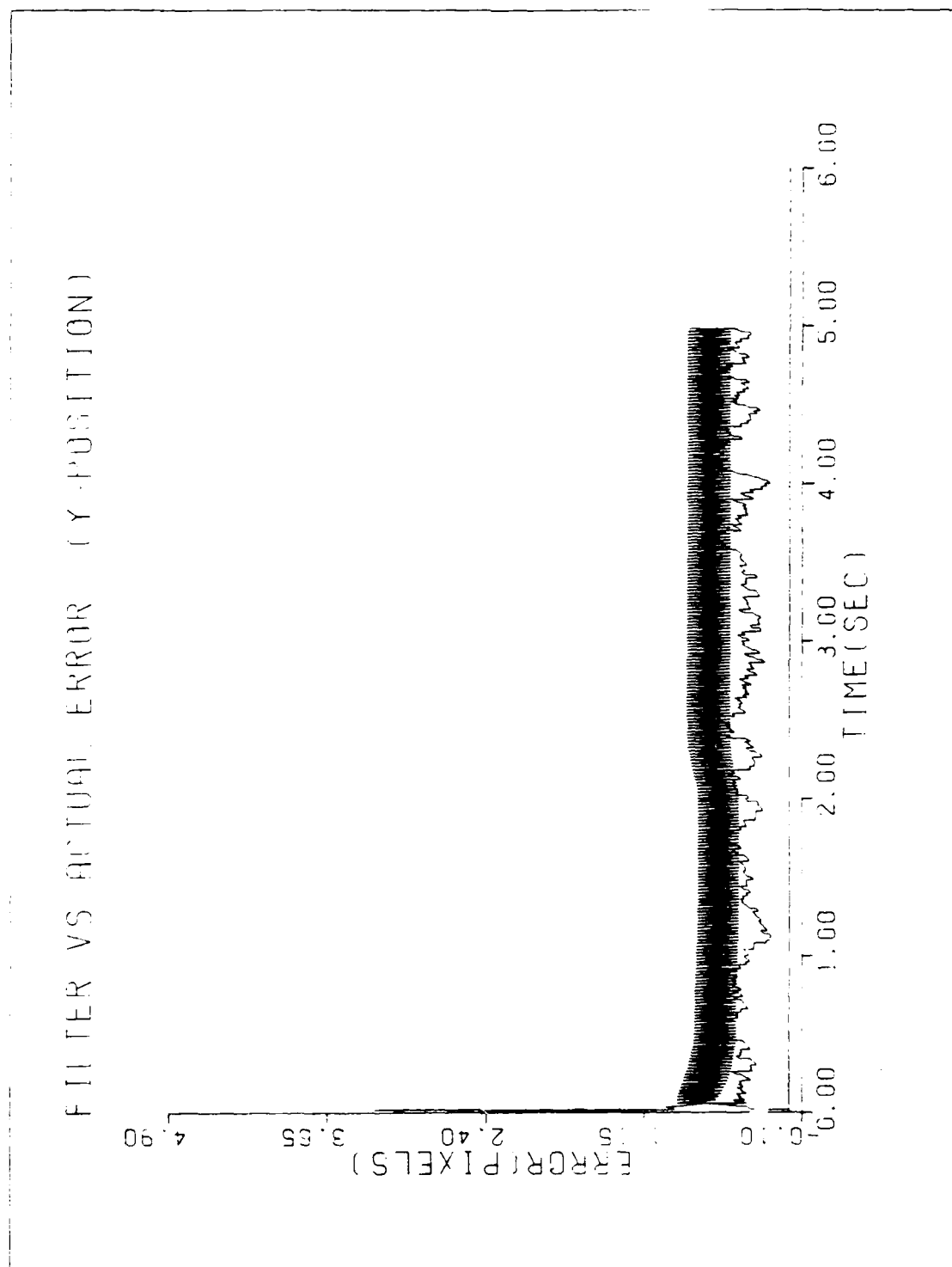


Figure C-10. GM/SNG-S/T2/10-G/50 Hz

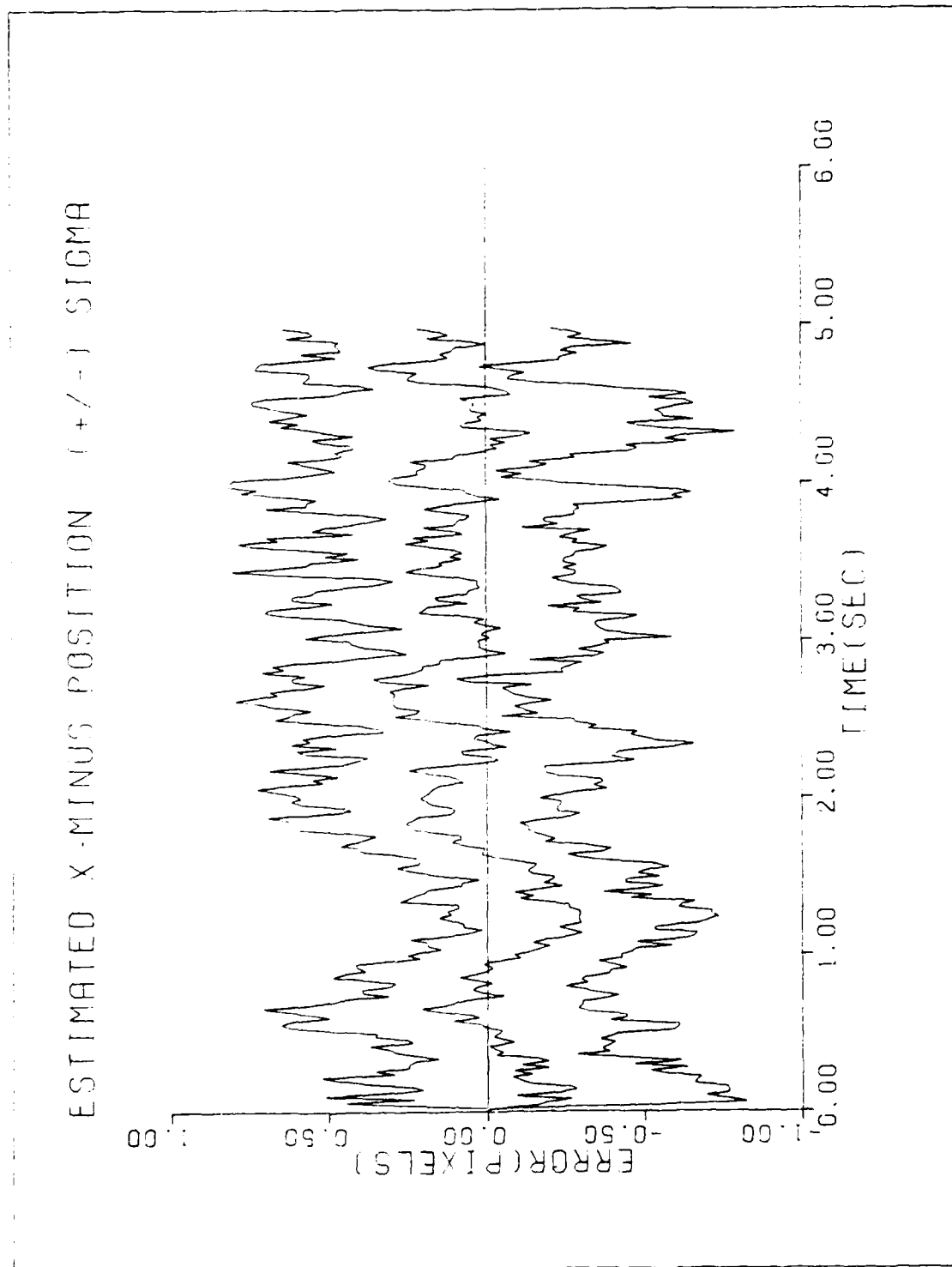


Figure C-11. GM/SNG-S/T2/10-G/50 Hz

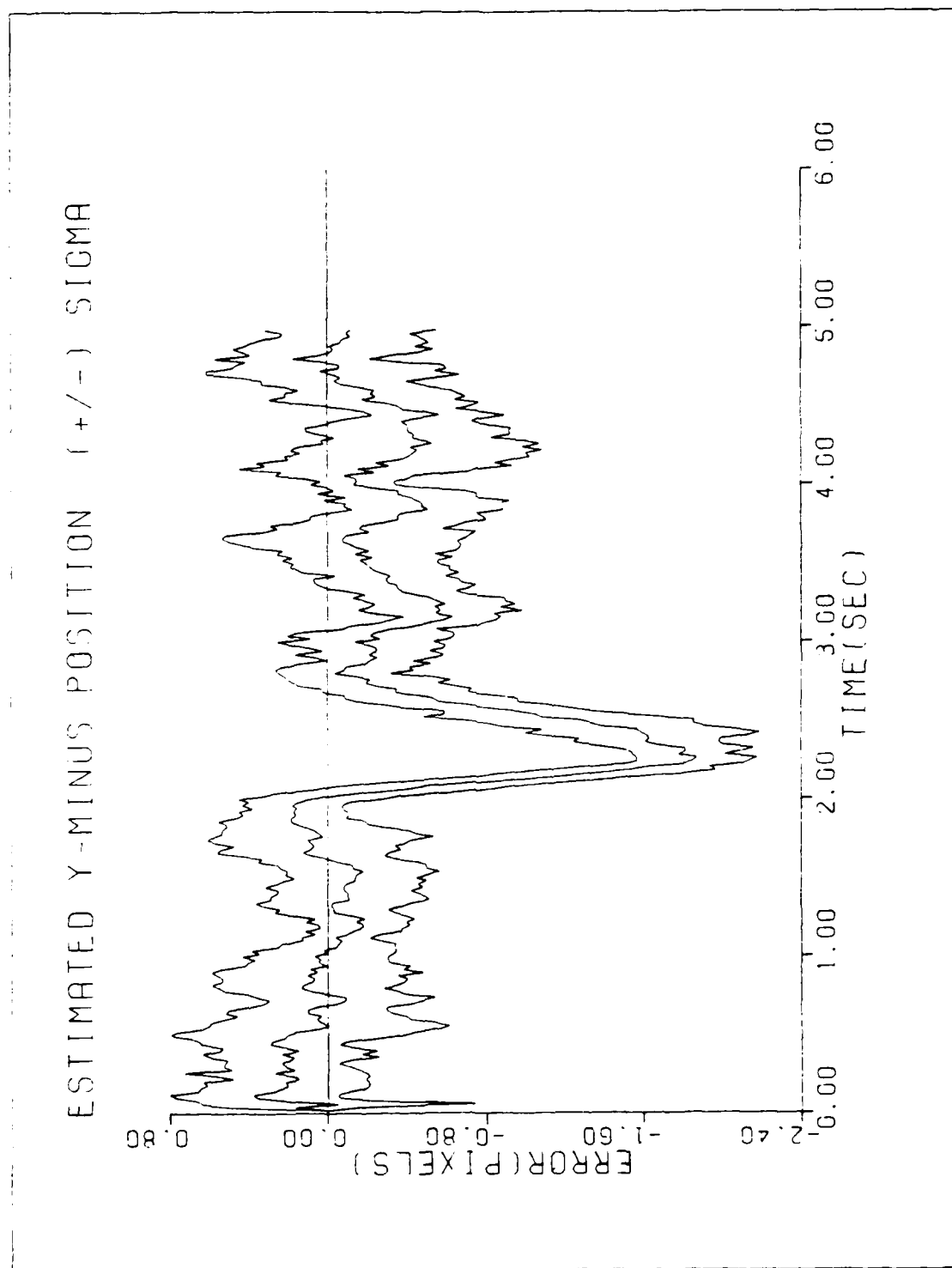


Figure C-12. GM/SNG-S/T2/10-G/50 Hz

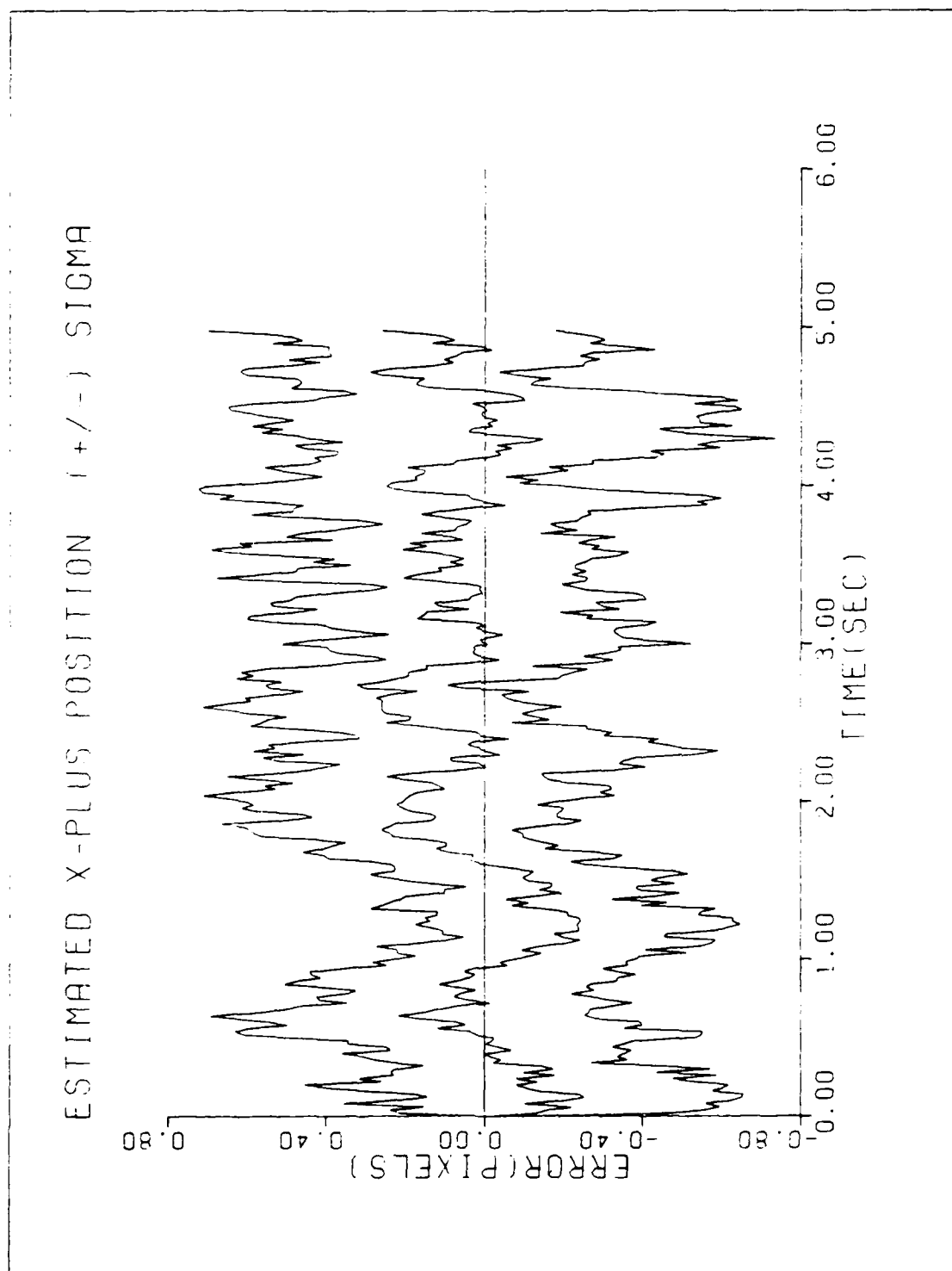


Figure C-13. GM/SNG-S/T2/10-G/50 Hz

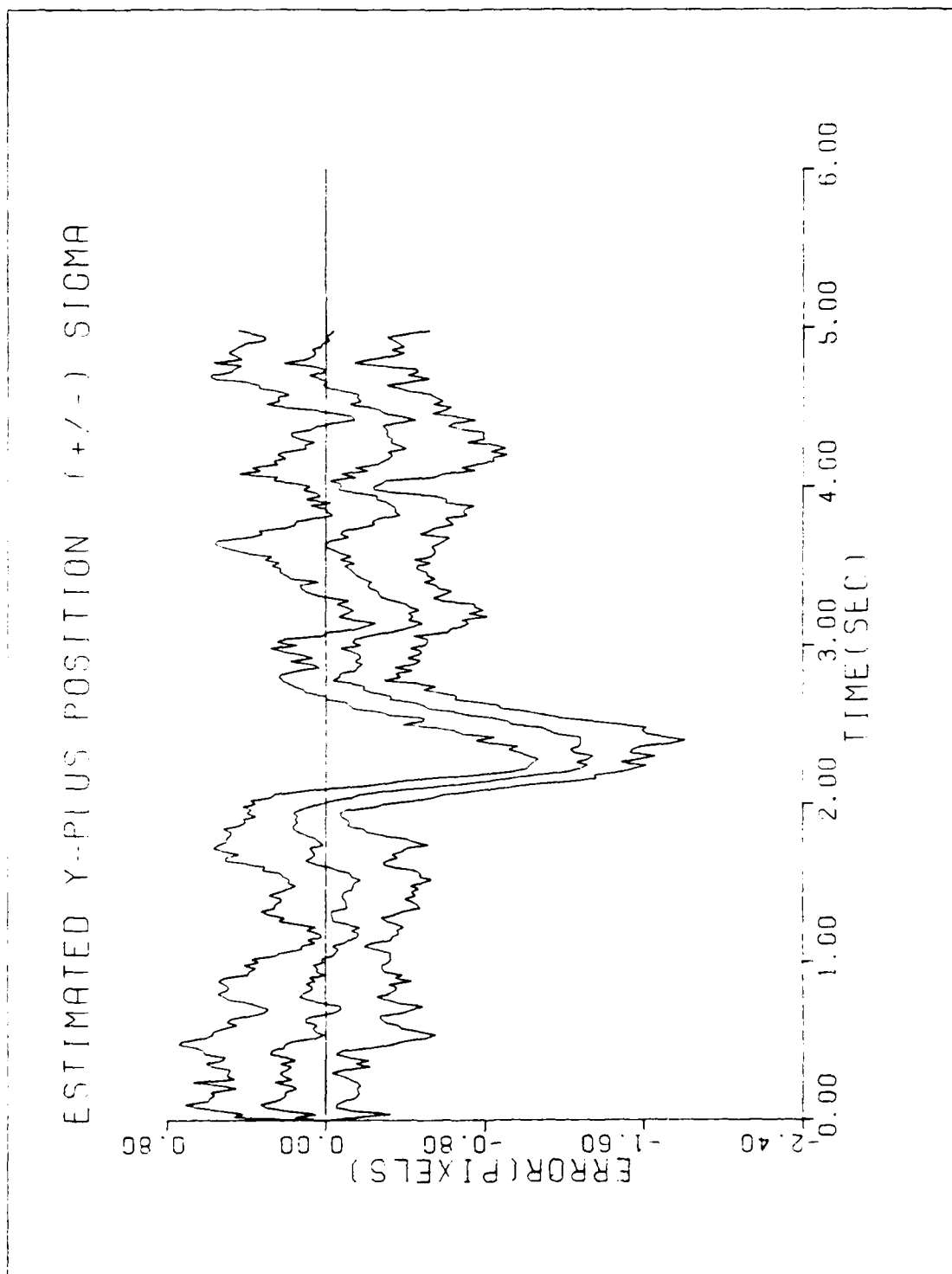


Figure C-14. GM/SNG-S/T2/10-G/50 Hz

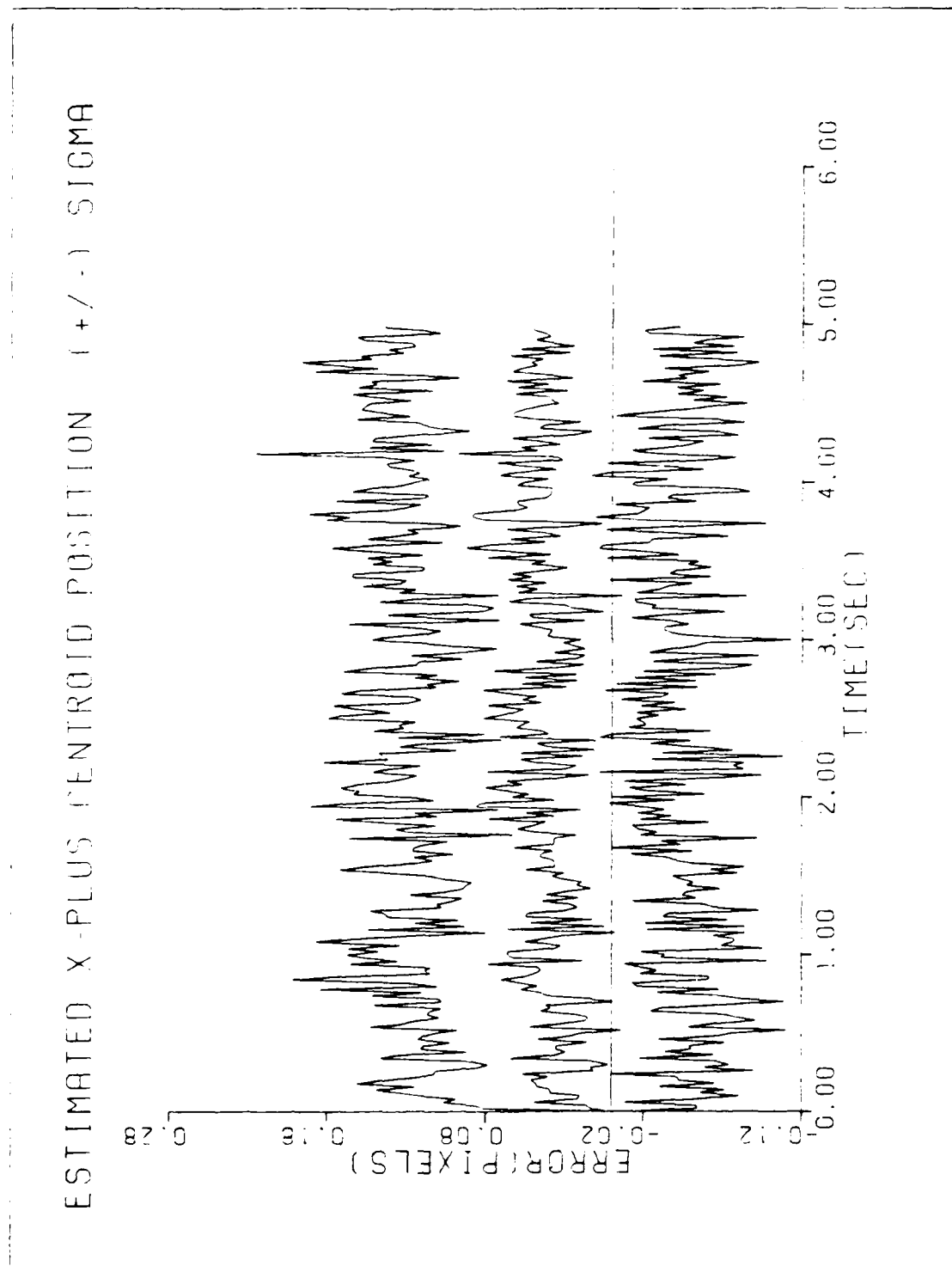


Figure C-15. GM/SNG-S/T2/10-G/50 Hz

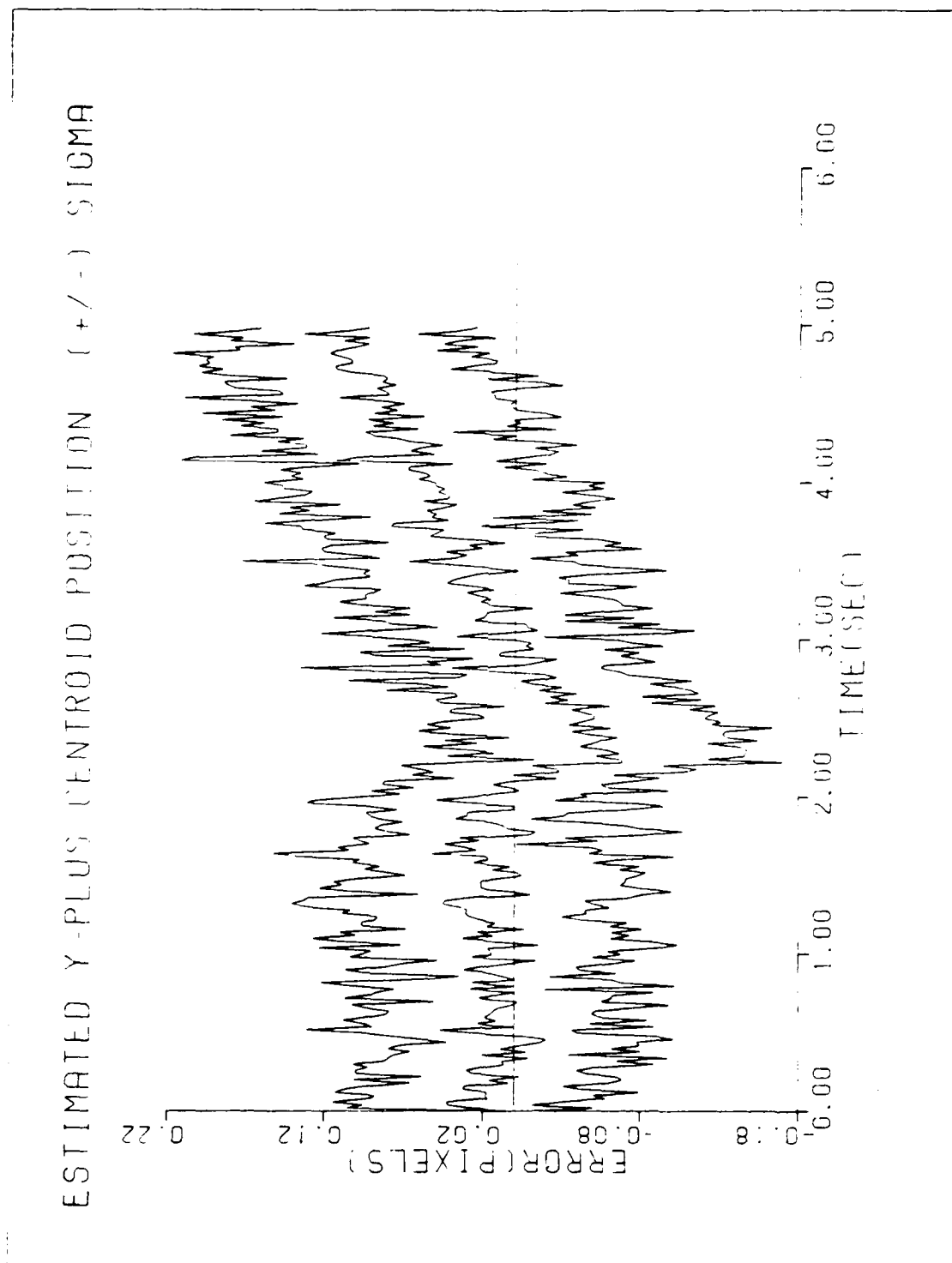


Figure C-16. GM/SNG-S/T2/10-G/50 Hz

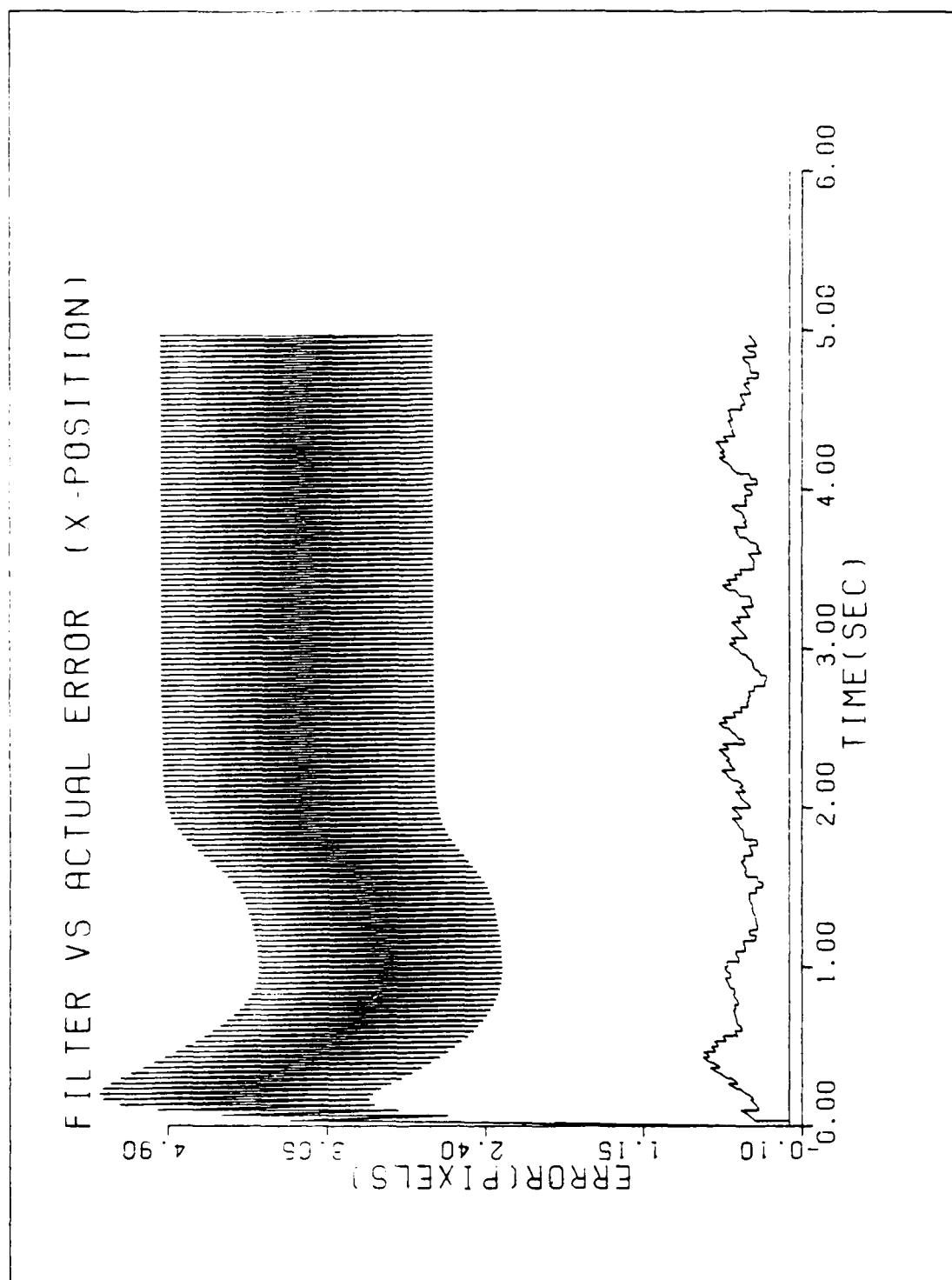


Figure C-17. CTR/SNG-S/T2/10-G

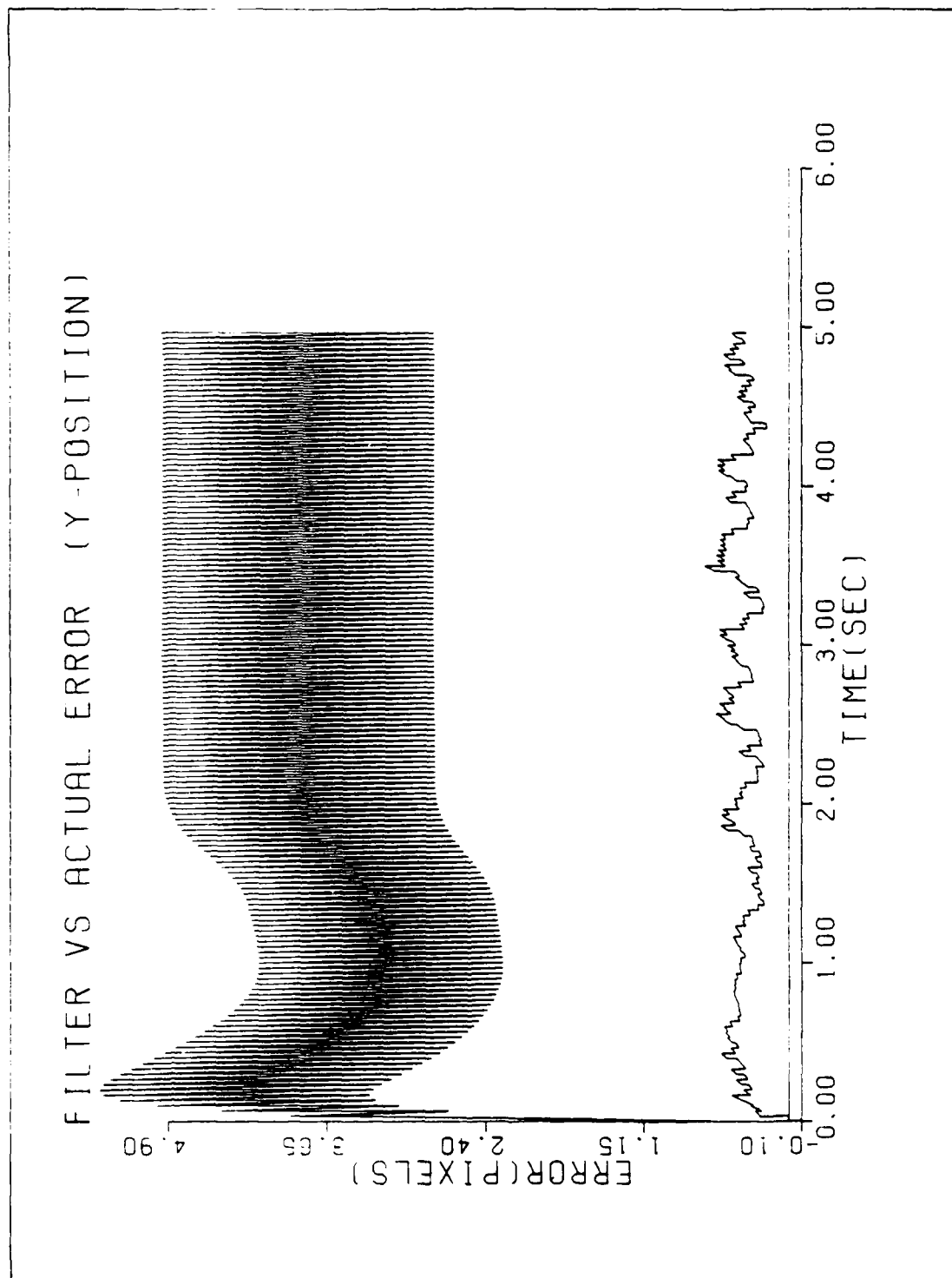


Figure C-18. CTR/SNG-S/T2/10-G

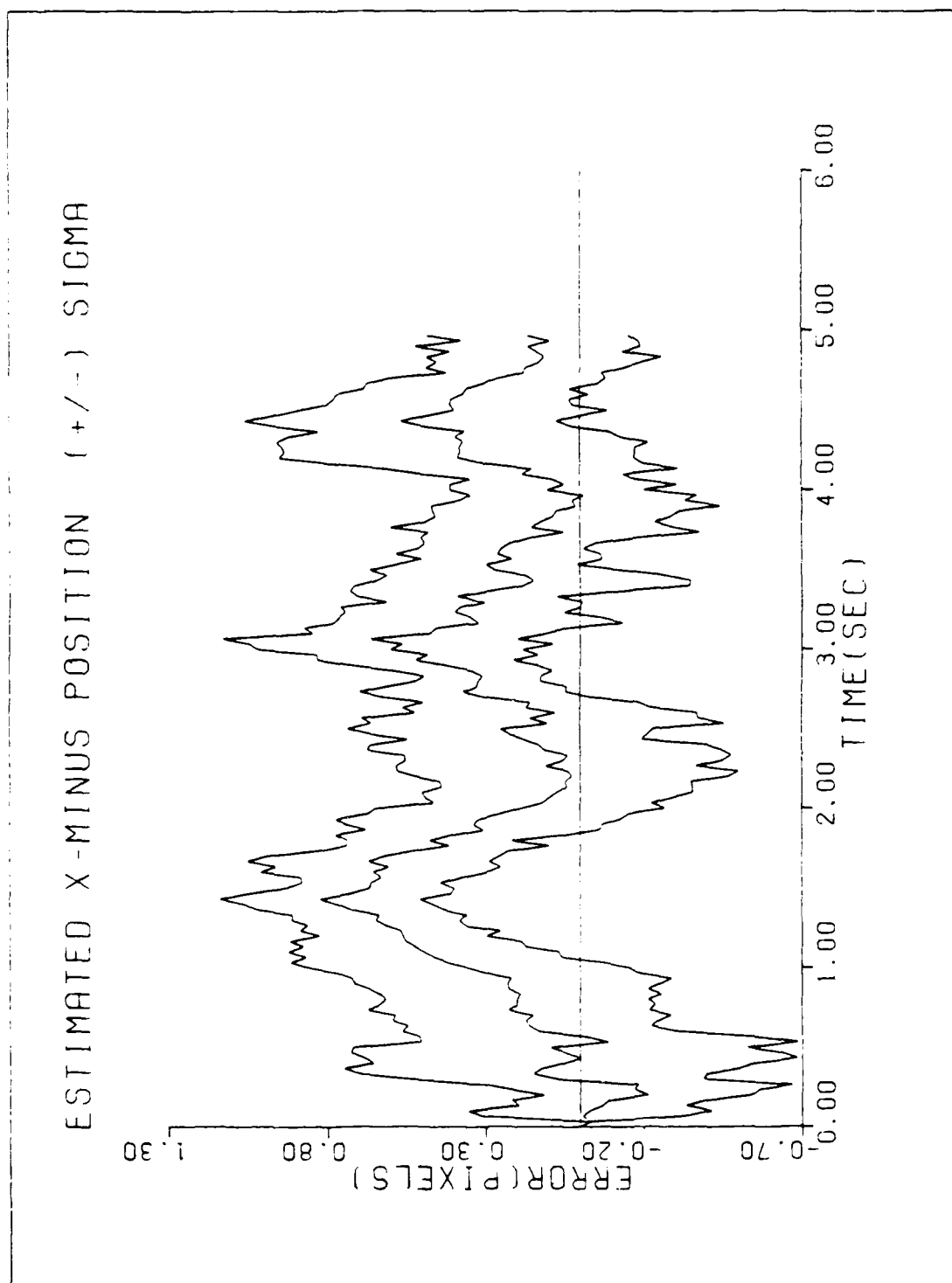


Figure C-19. CTR/SNG-S/T2/10-G

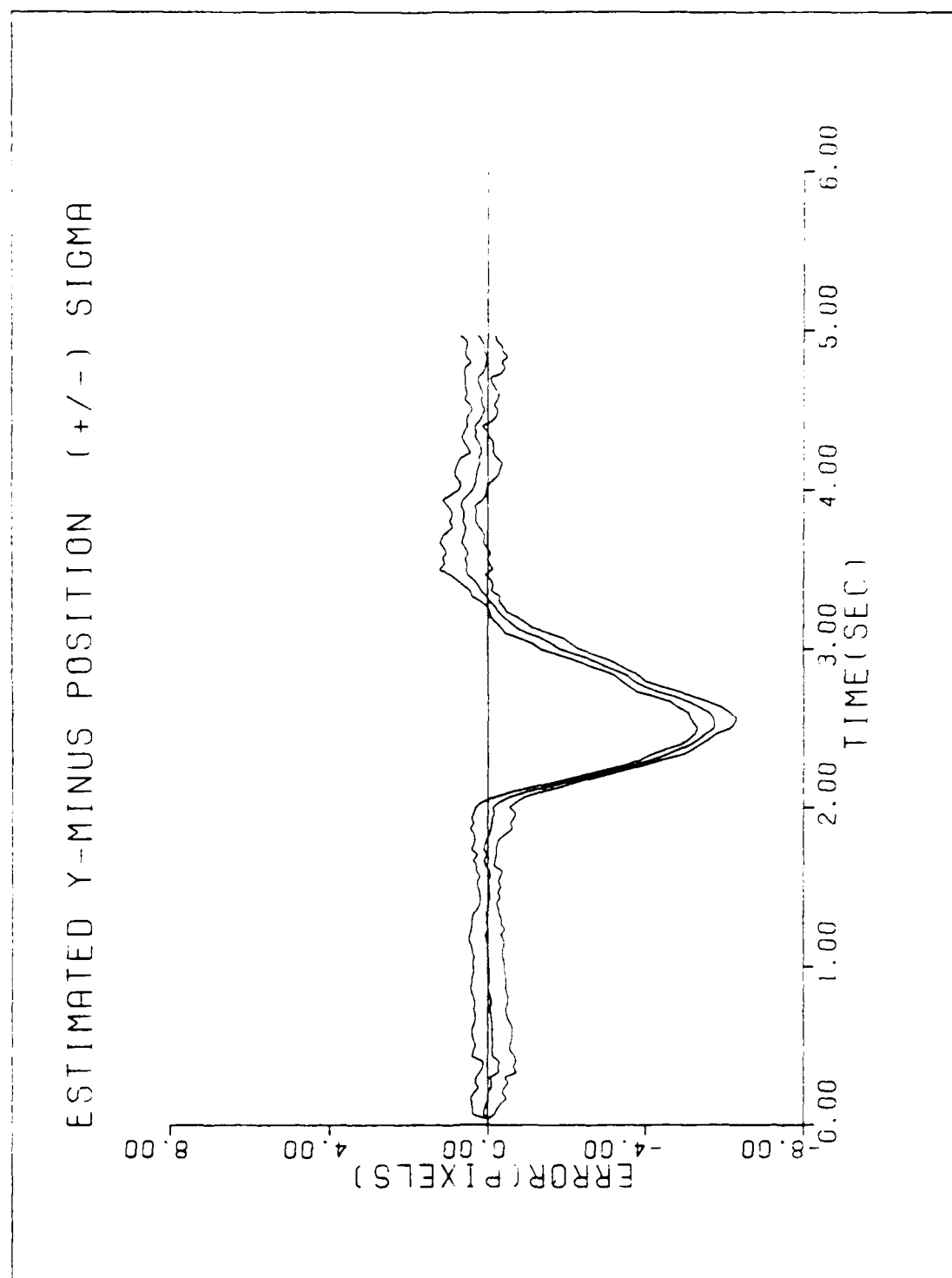


Figure C-20. CTR/SNG-S/T2/10-G

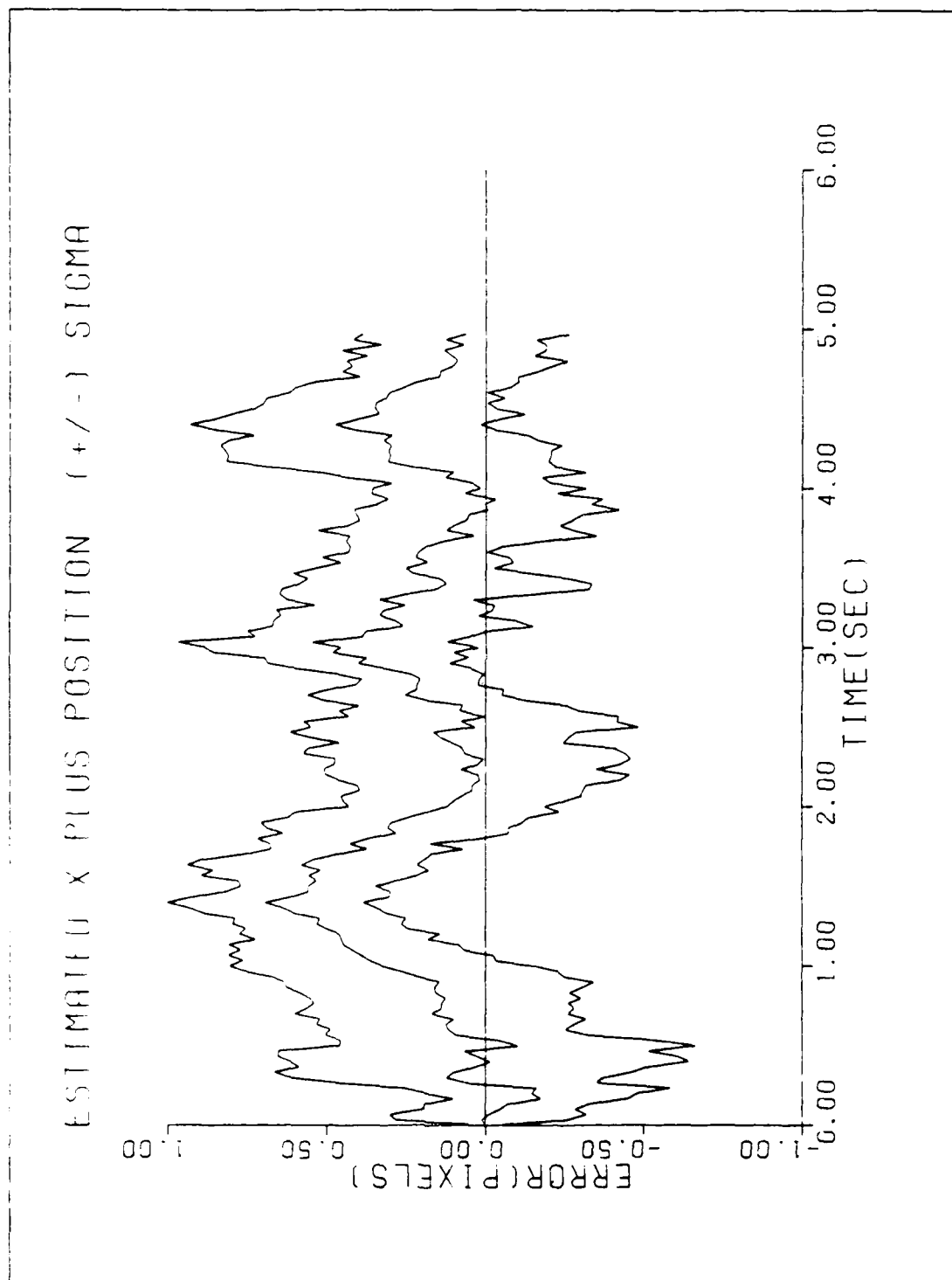


Figure C-21. CTR/SNG-S/T2/10-G

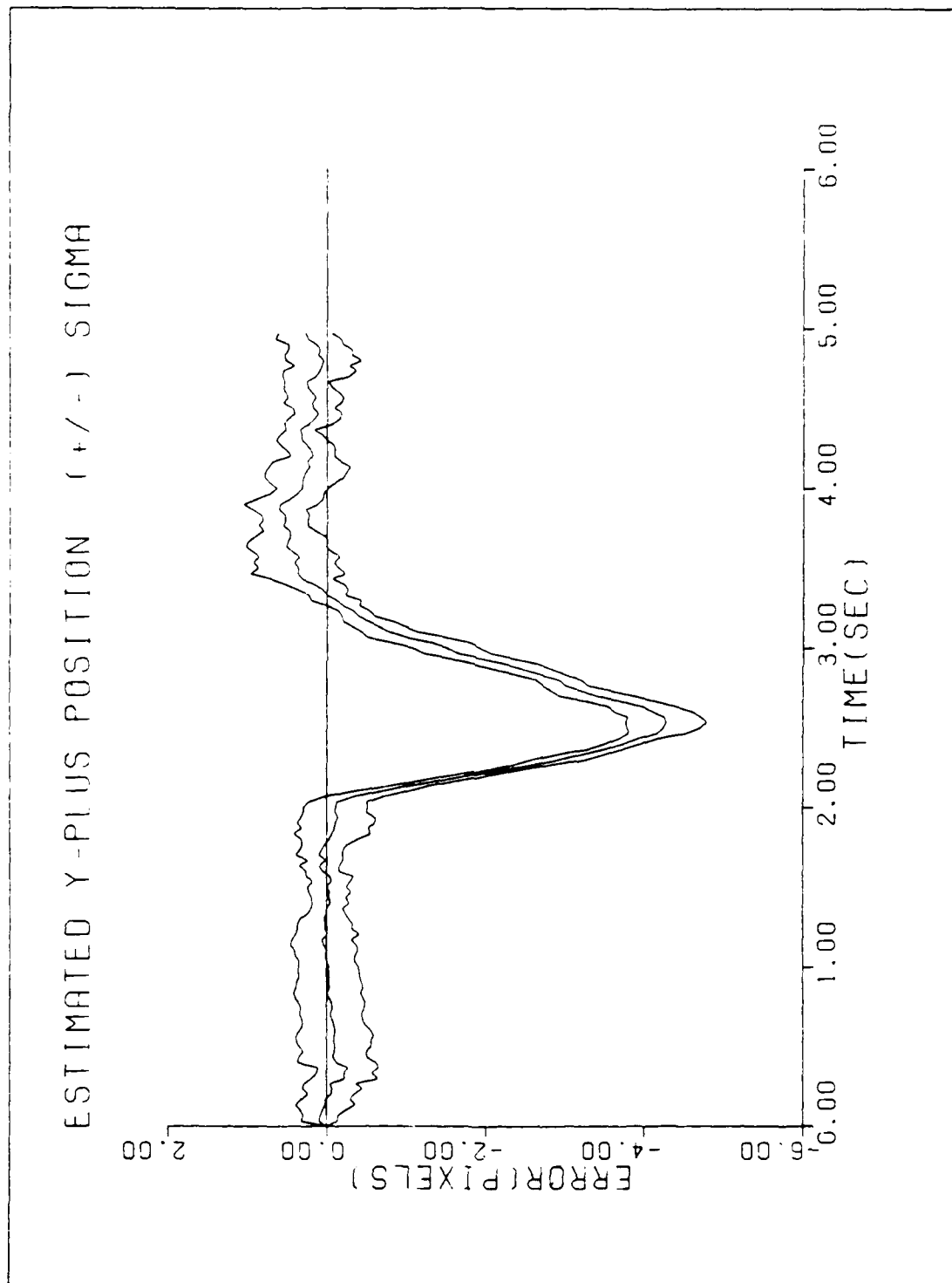


Figure C-22. CTR/SNG-S/T2/10-G

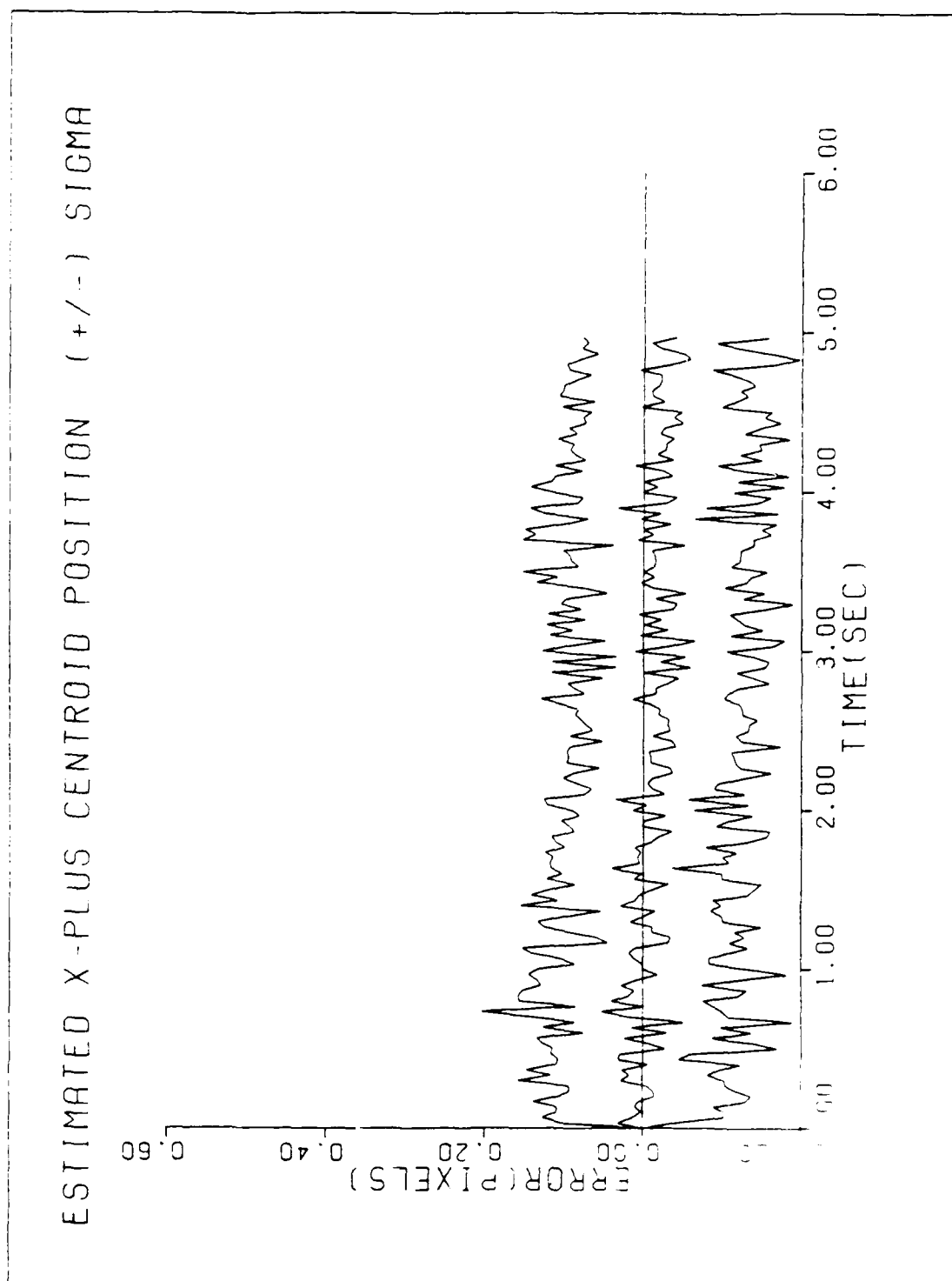


Figure C-23. CTR/SNG-5/12.1

AD-A190 619

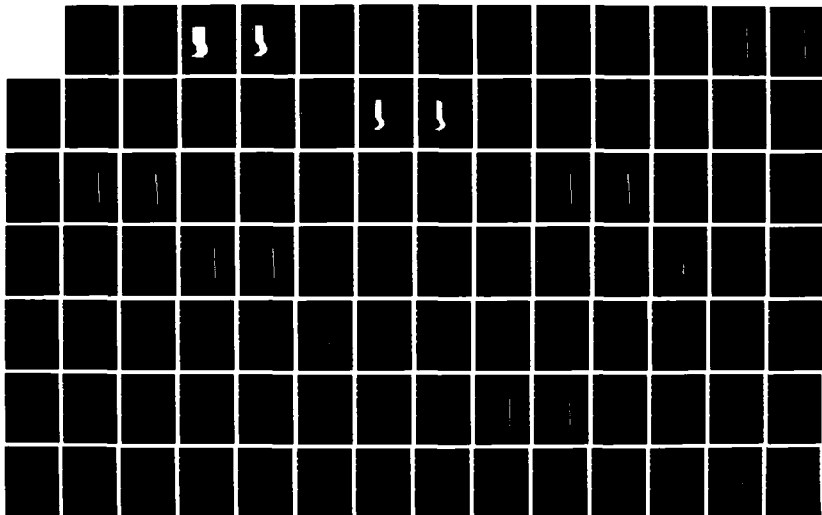
A MULTIPLE MODEL ADAPTIVE TRACKING ALGORITHM AGAINST
AIRBORNE TARGETS(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING
T A LEENEY DEC 87 AFIT/GE/ENG/87D-37

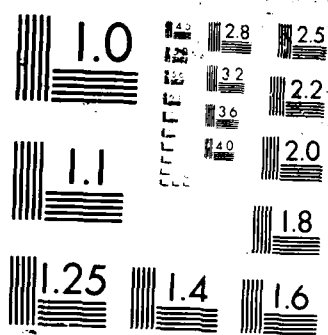
3/4

UNCLASSIFIED

F/G 17/3.1

NL





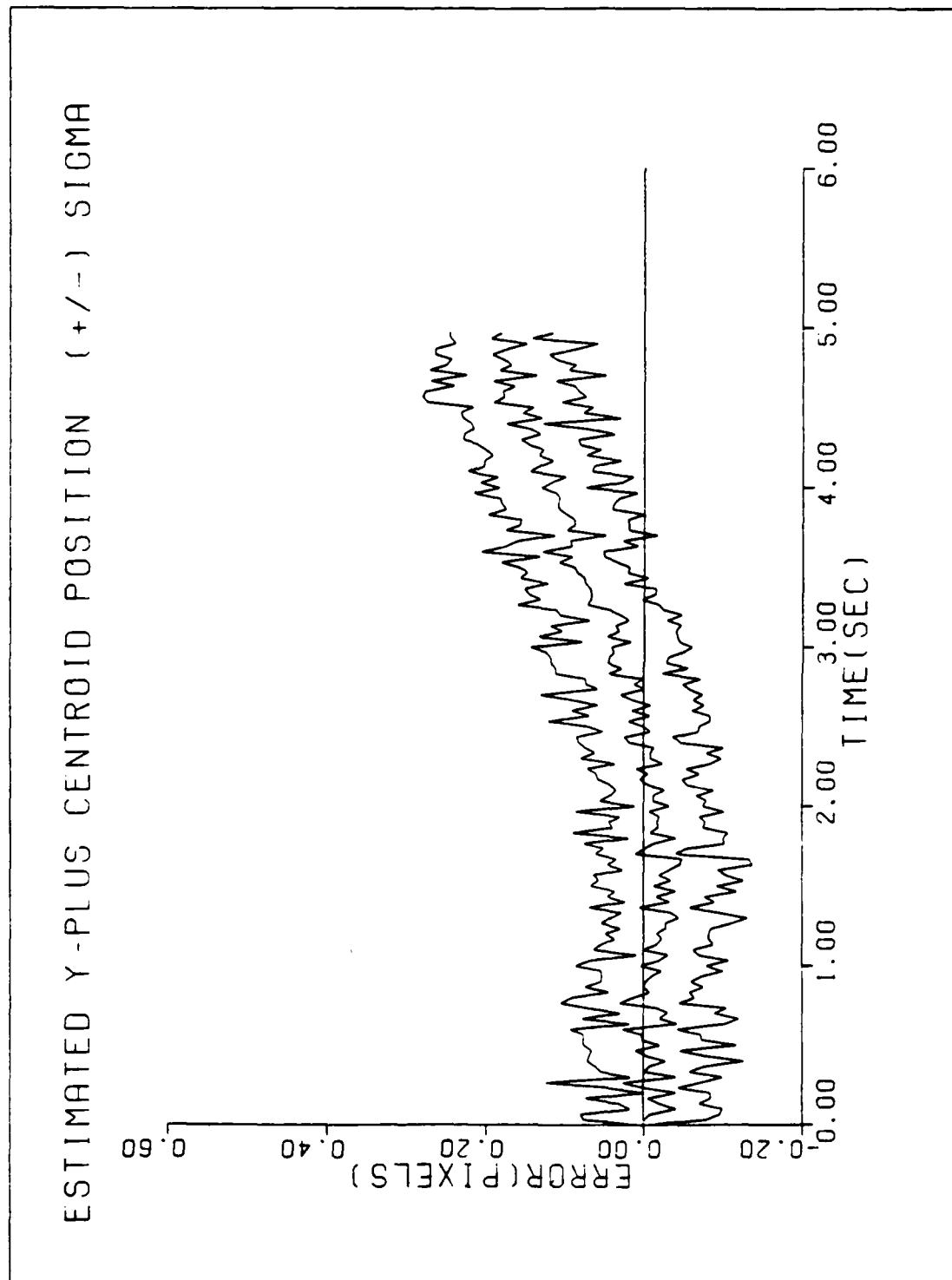


Figure C-24. CTR/SNG-S/T2/10-6

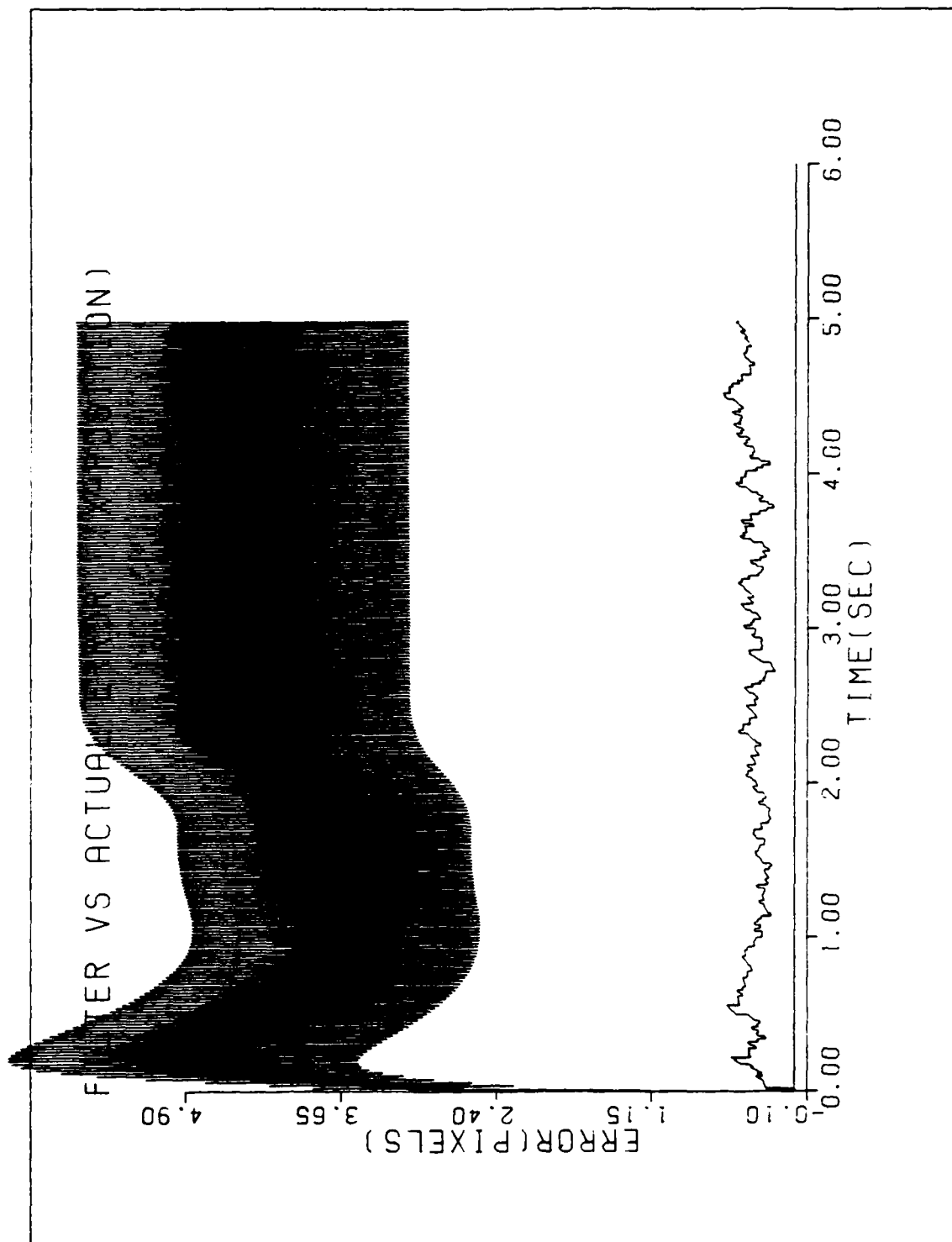


Figure C-25. CTR/SNG-S/T2/10-G/50 Hz

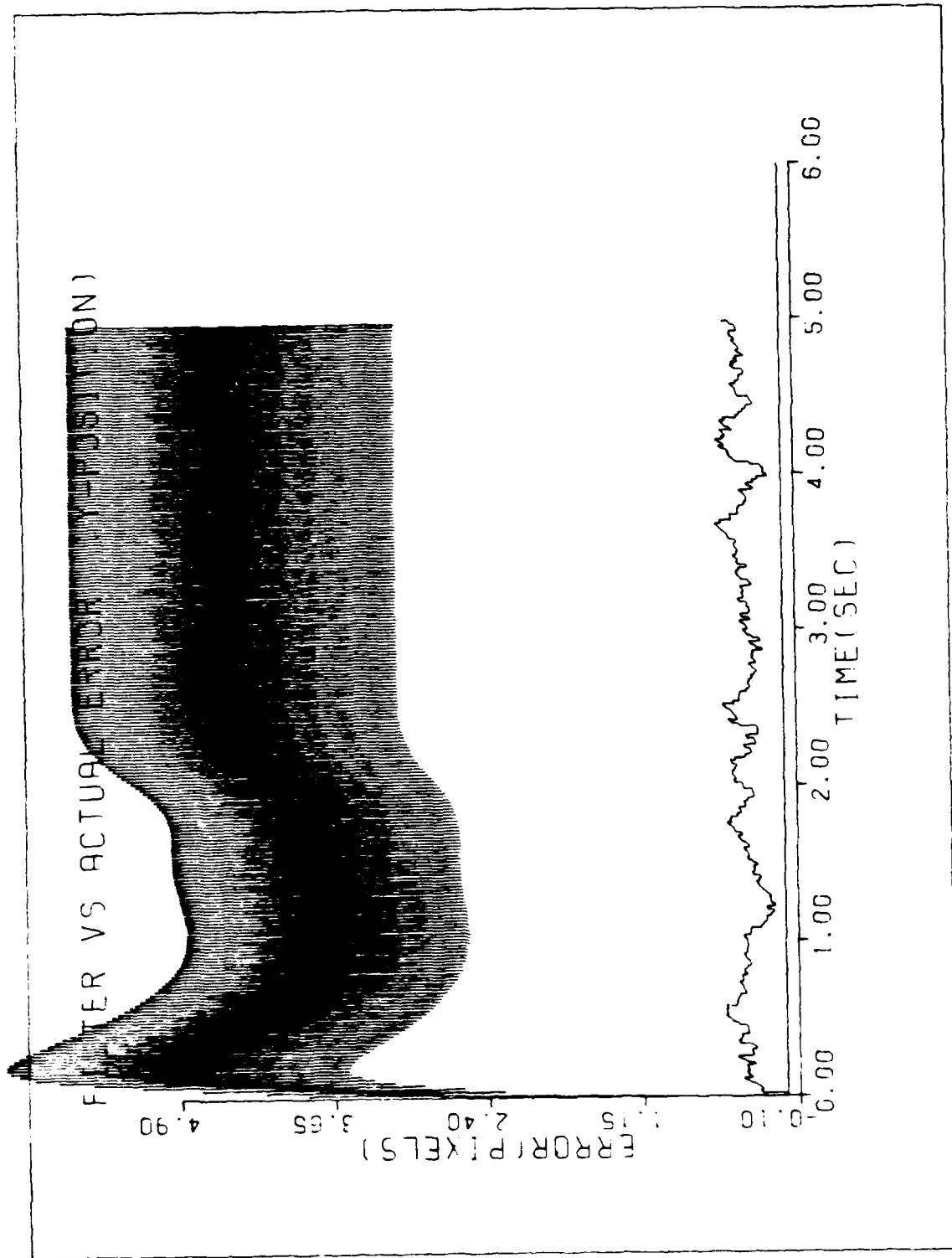


Figure C-26. CTR/SNG-S/T2/10-G/50 Hz

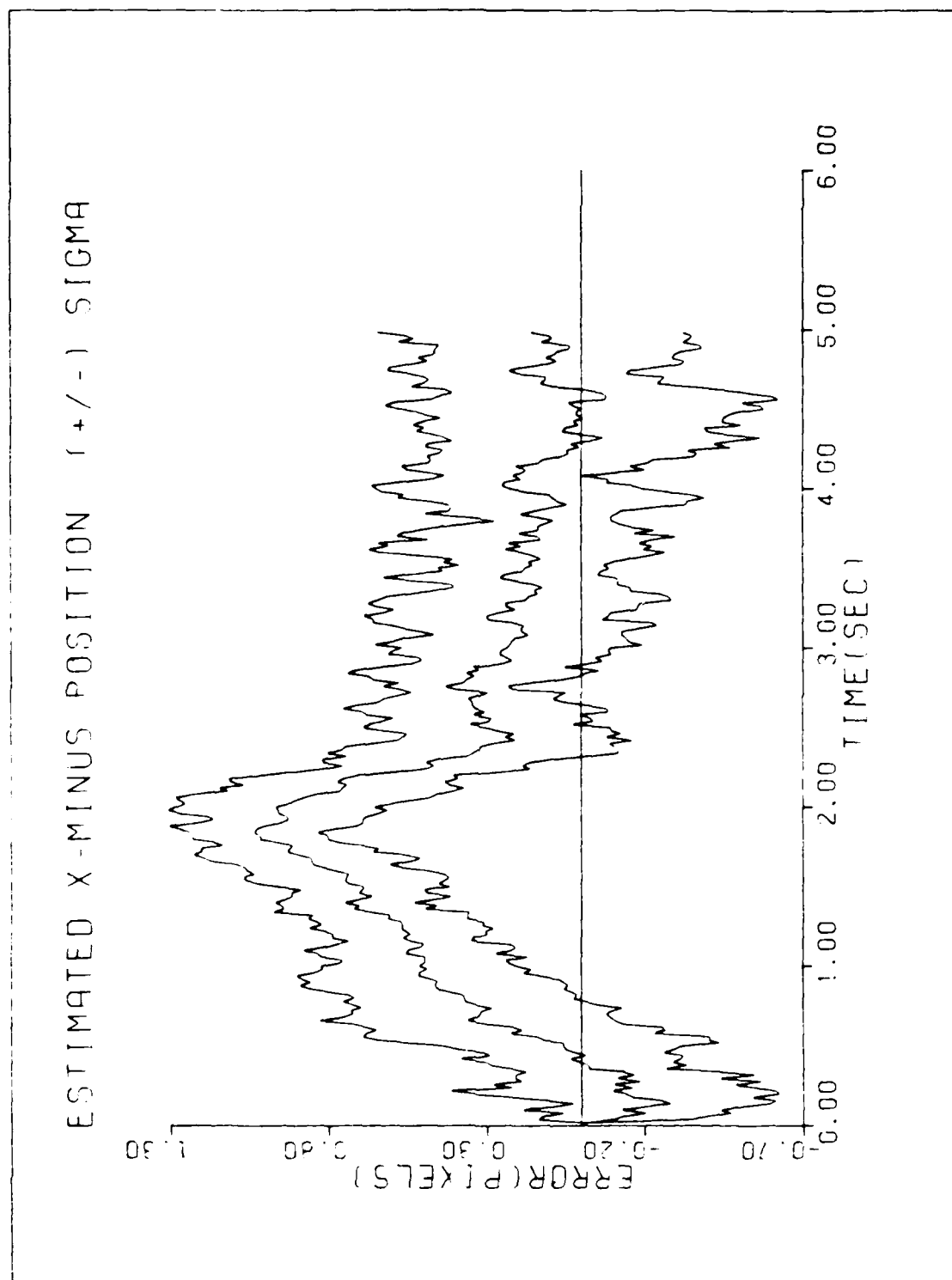


Figure C-27. CTR/SNG-S/T2/10-G/50 Hz

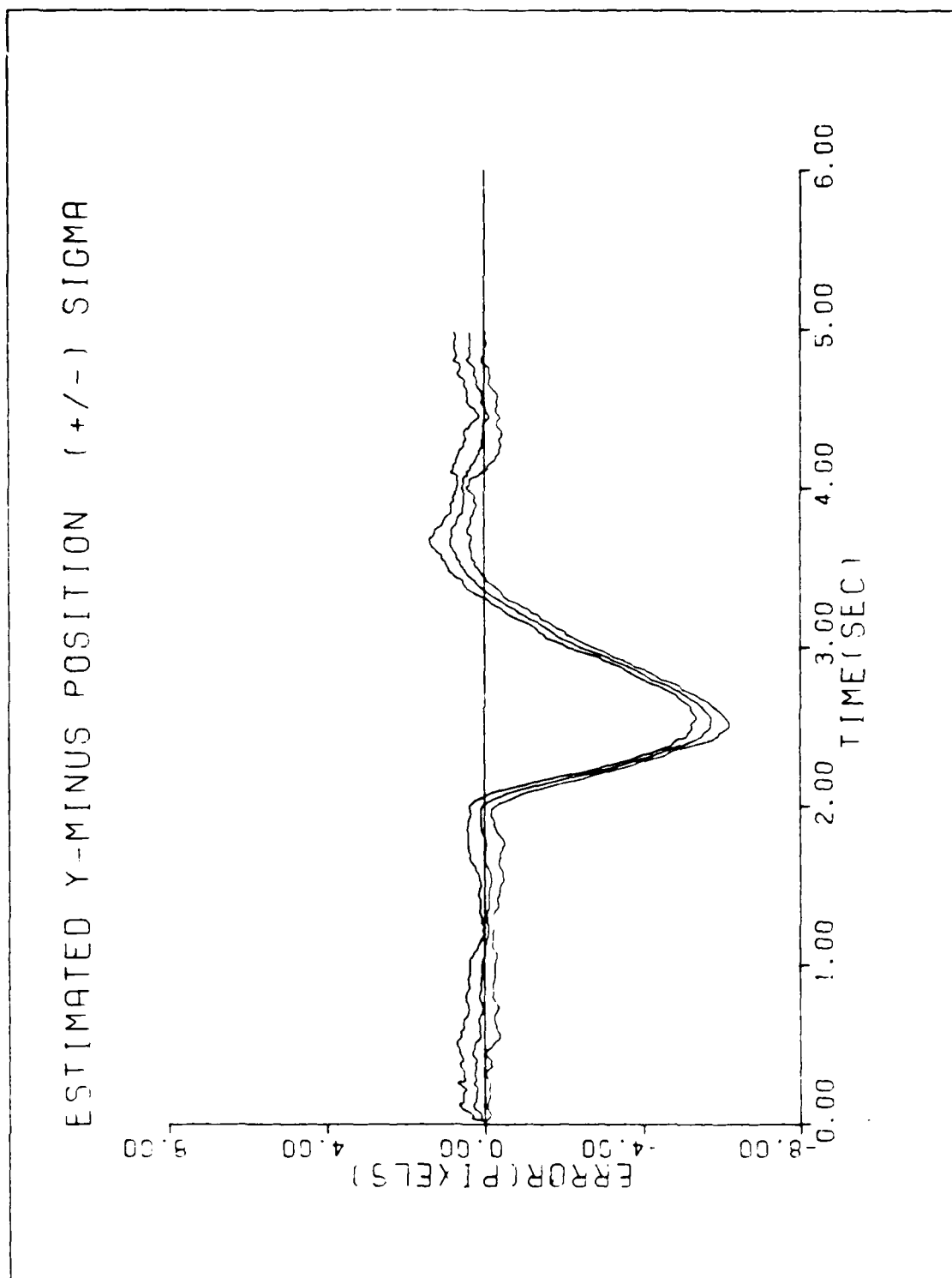


Figure C-28. CTR/SNG-S/T2/10-G/50 Hz

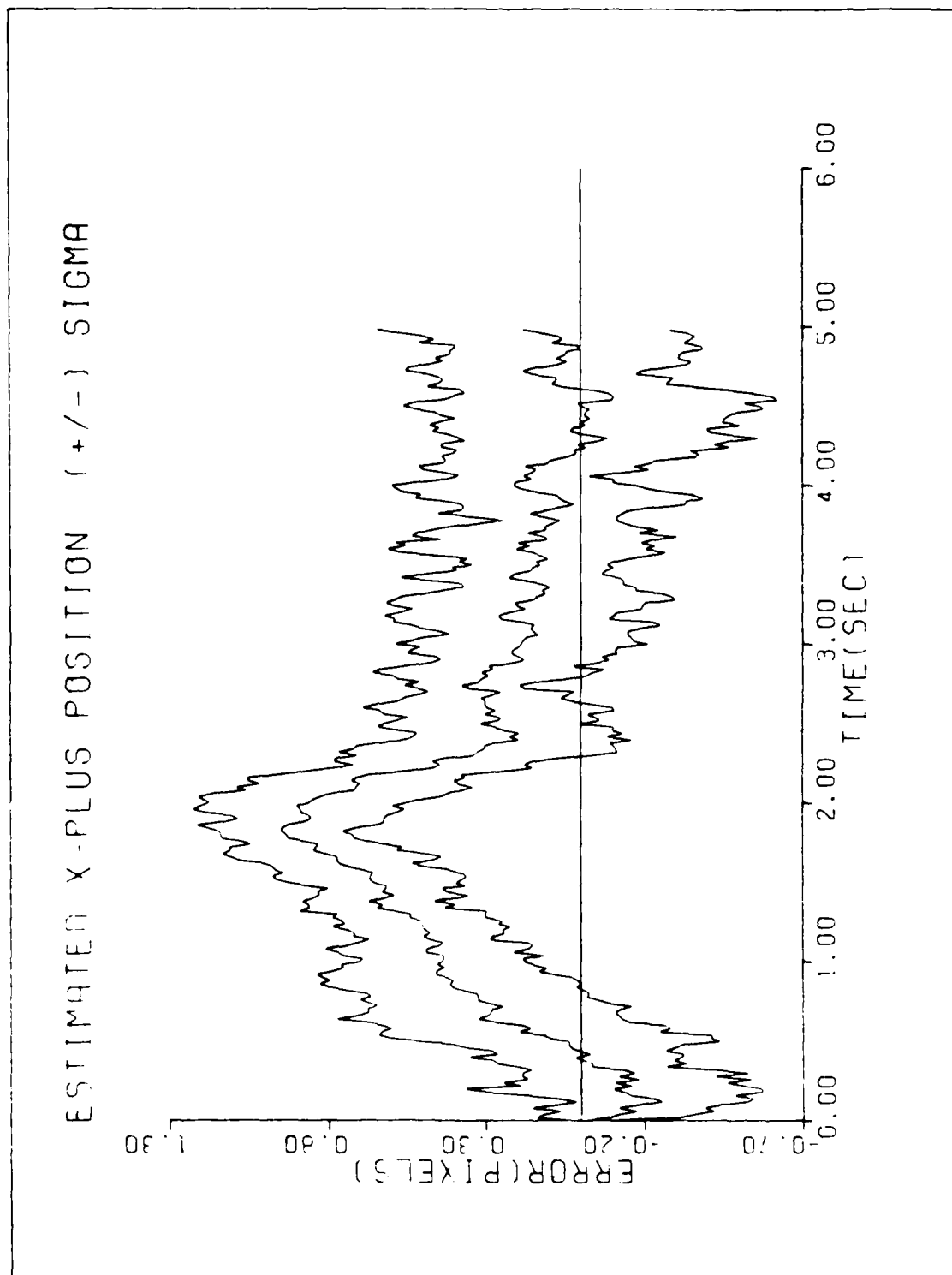


Figure C-29. CTR/SNG-S/T2/10-G/50 Hz

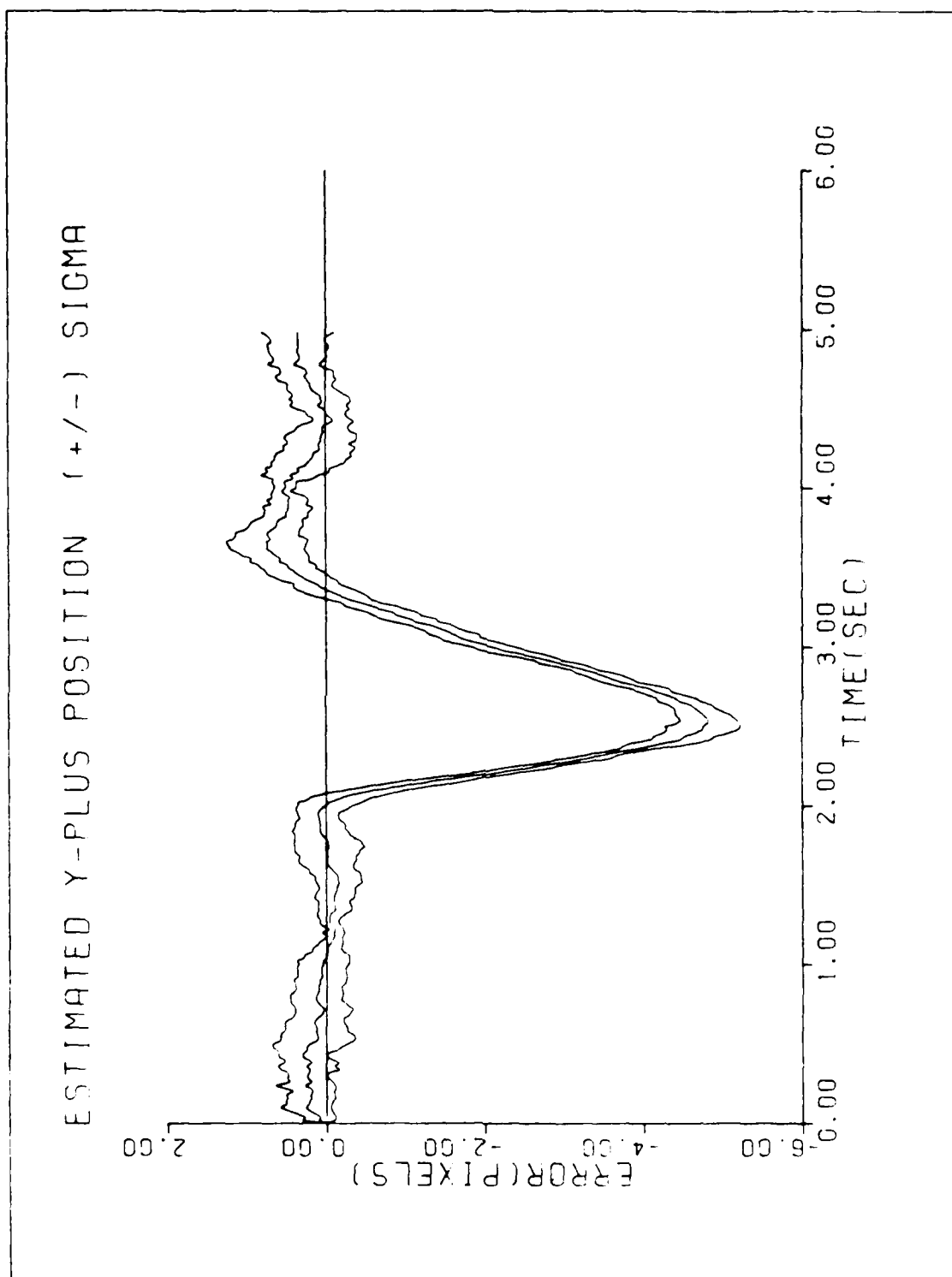


Figure C-30. CTR/SNG-S/T2/10-G/50 Hz

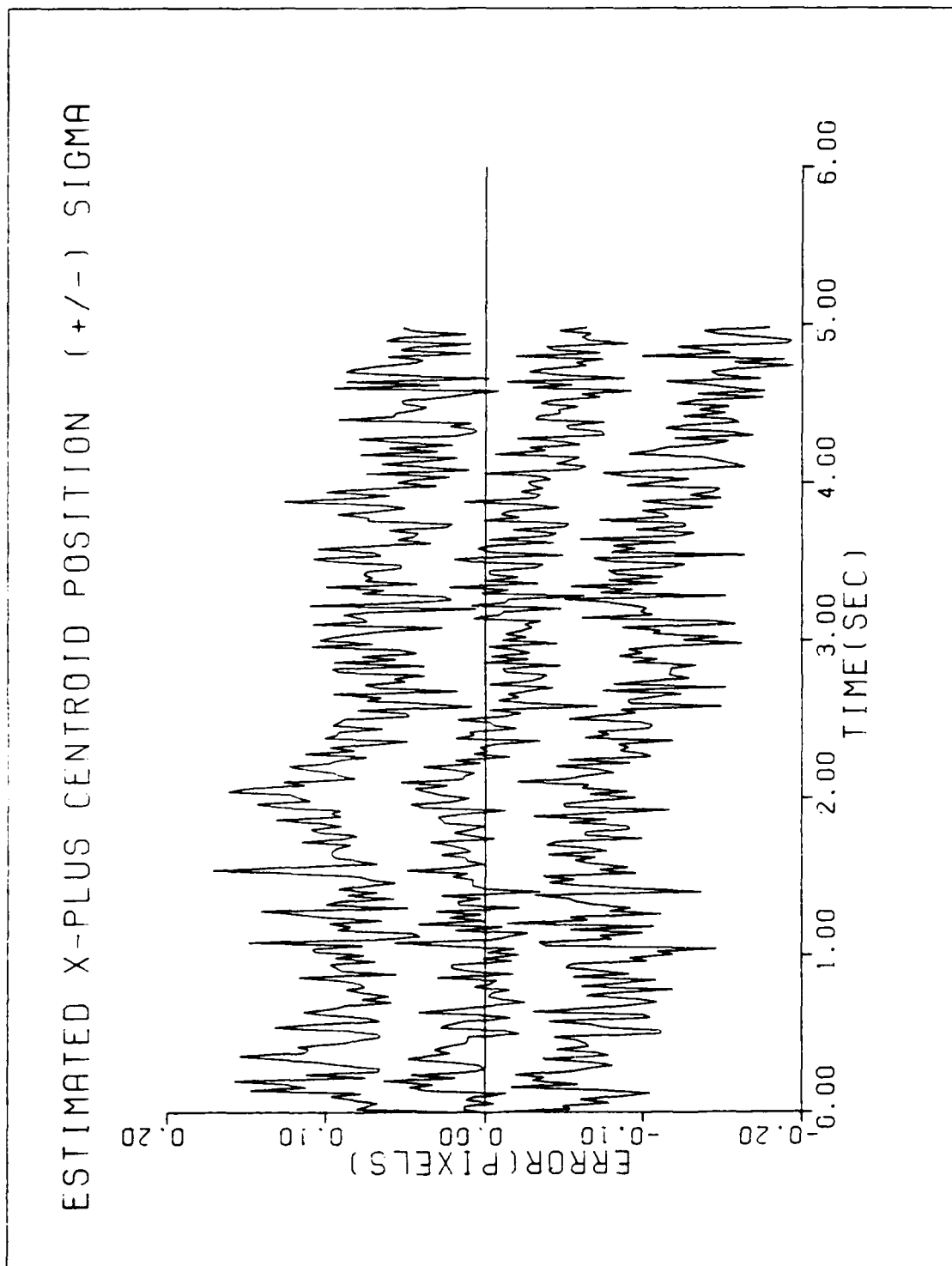


Figure C-31. CTR/SNG-S/T2/10-G/50 Hz

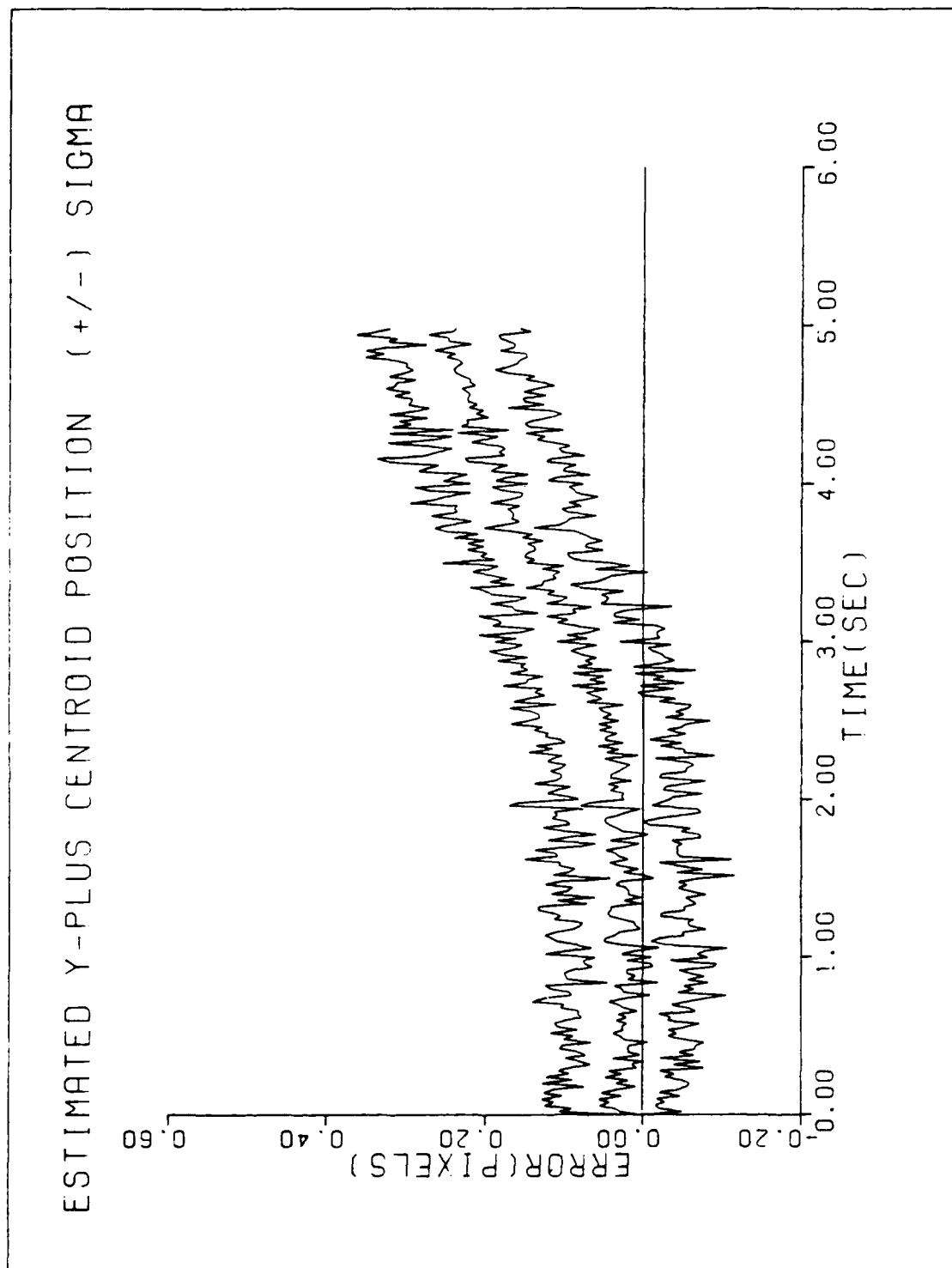


Figure C-32. CTR/SNG-S/T2/10-G/50 Hz

APPENDIX D

**SINGLE FILTER SIMULATIONS
BENDING AND VIBRATION ANALYSIS
(REFERENCED FROM SECTION 6.3)**

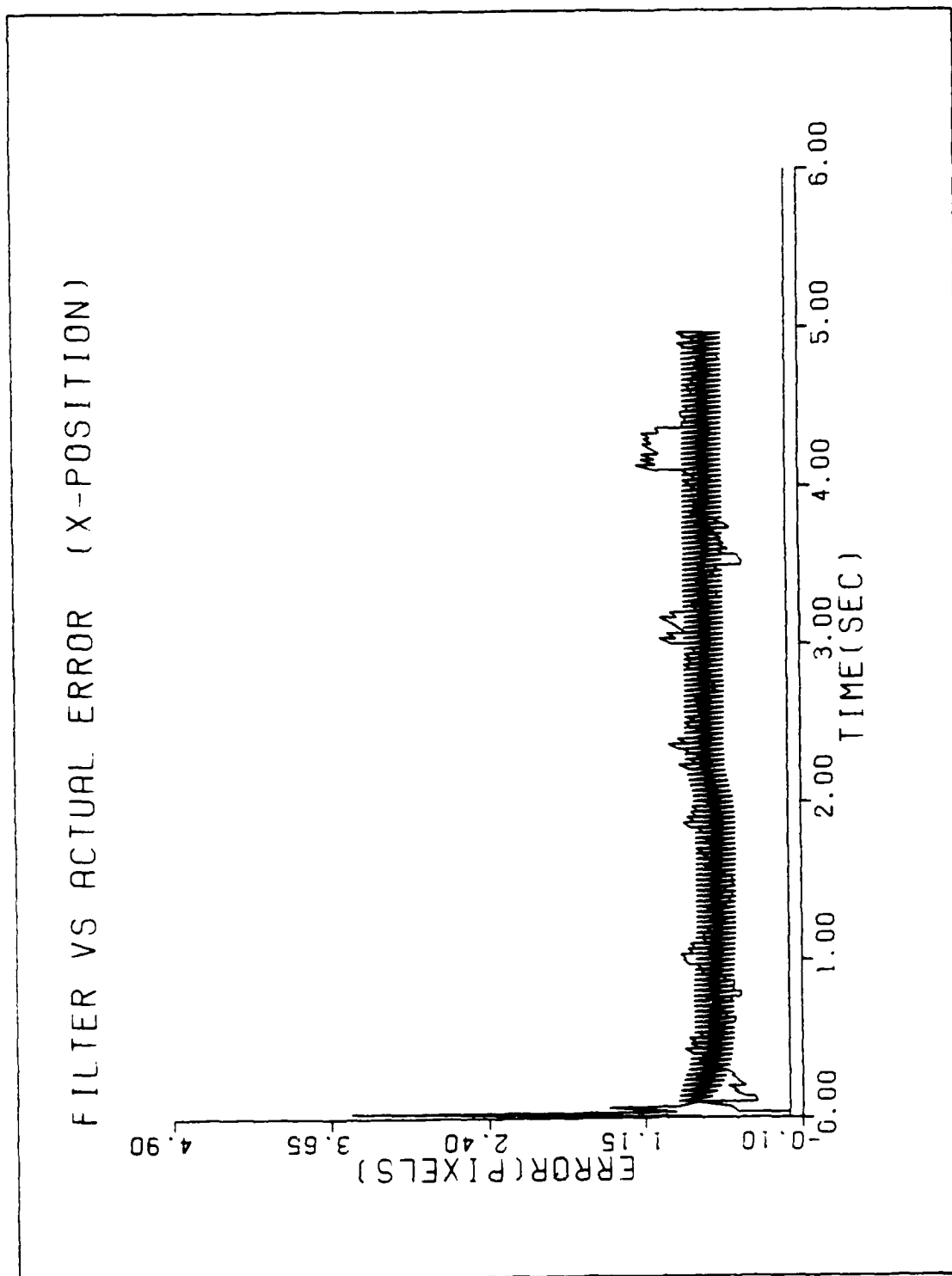


Figure D-1. GM/SNG-S/T2/10-G/QB1

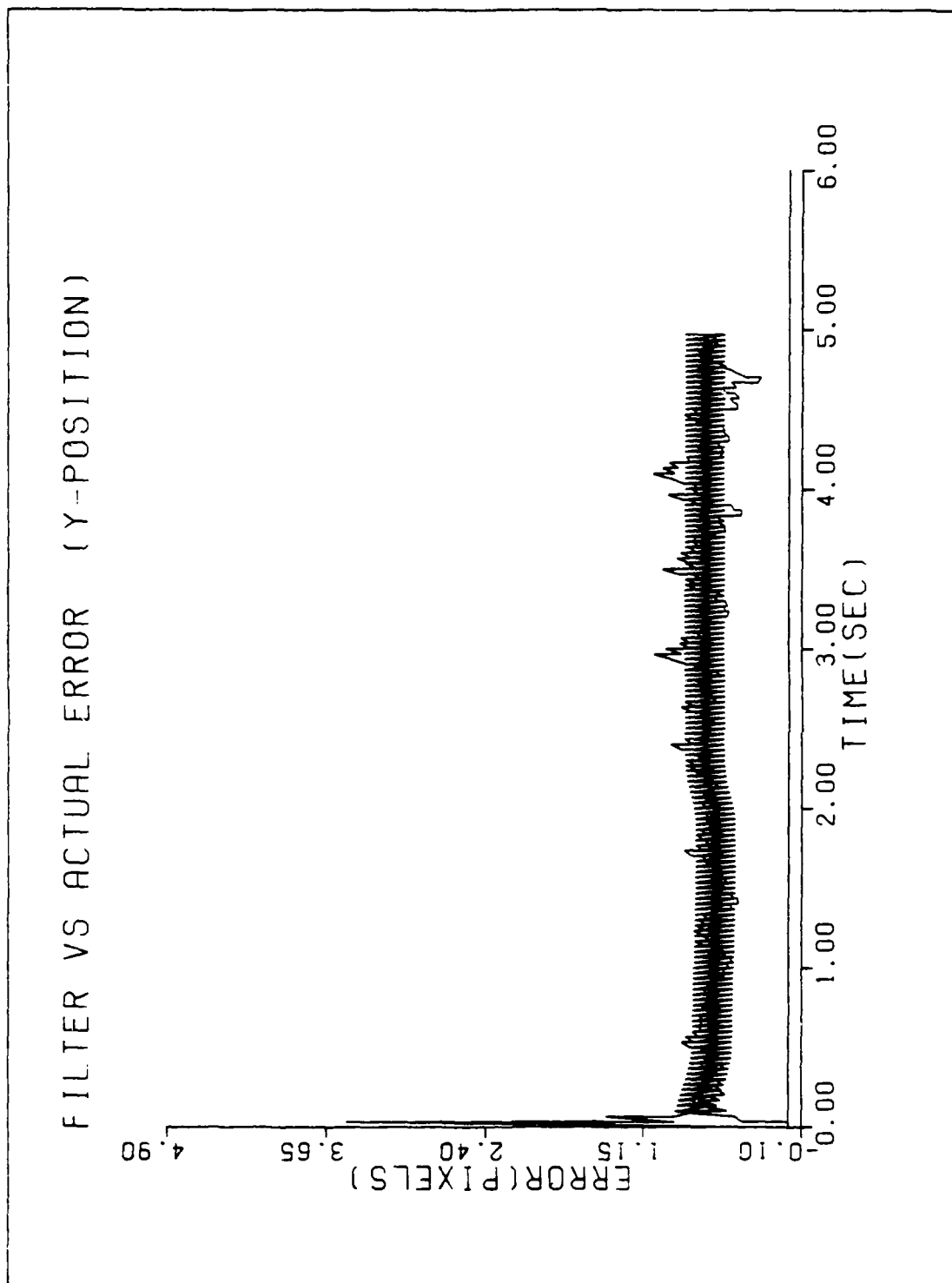


Figure D-2. GM/SNG-S/T2/10-G/QB1

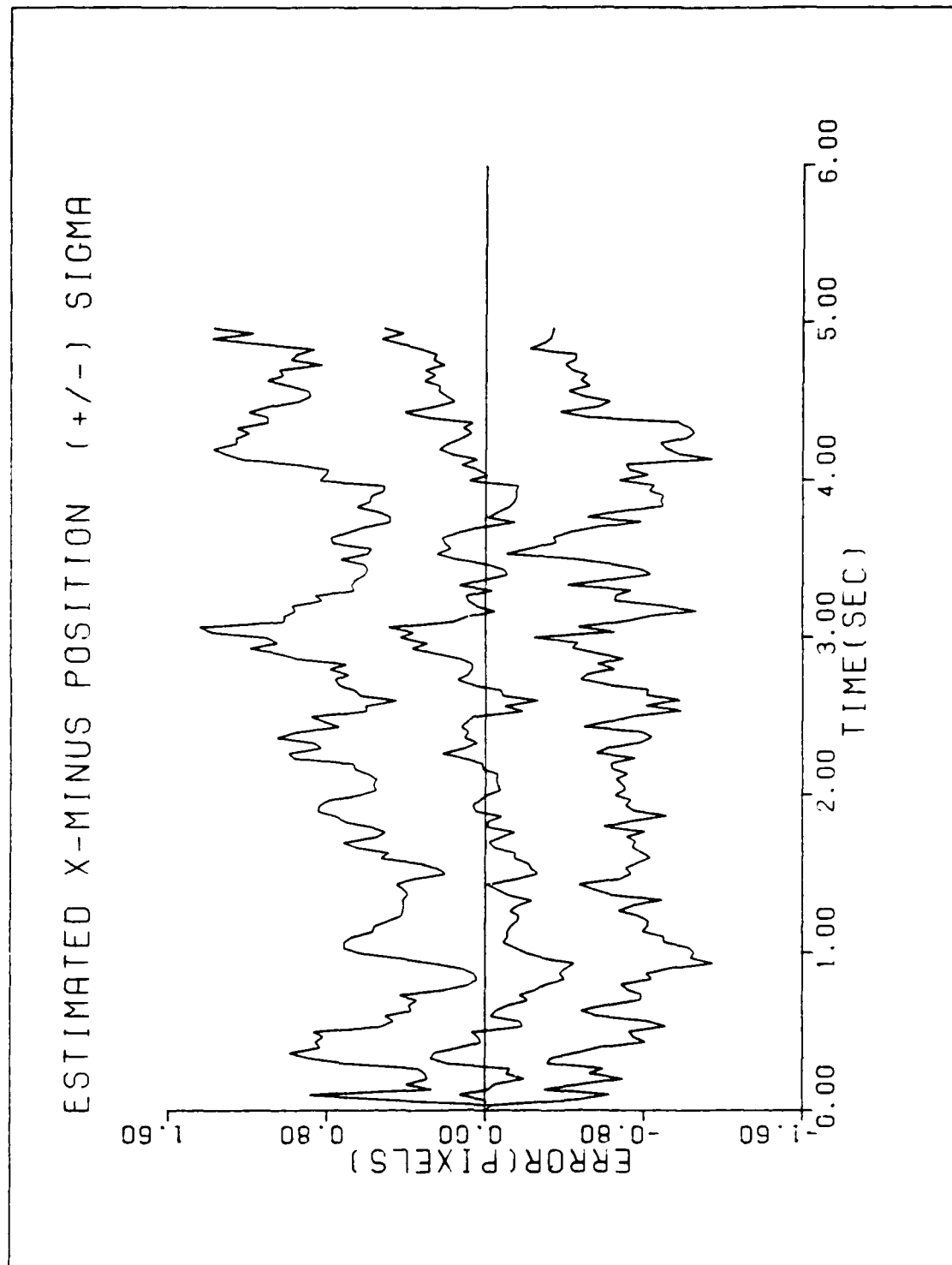


Figure D-3. GM/SNG-S/T2/10-G/QB1

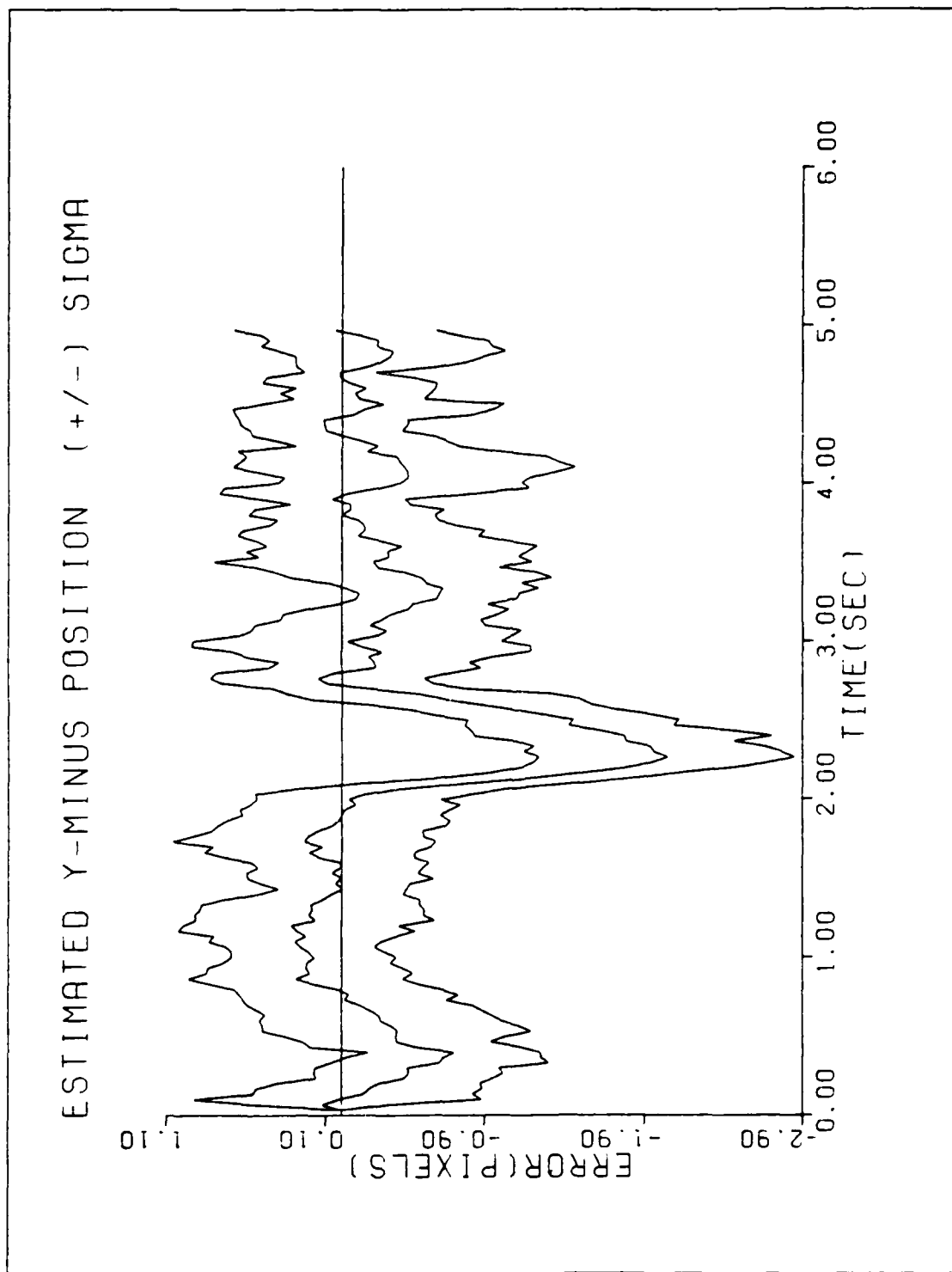


Figure D-4. GM/SNG-S/T2/10-G/QB1

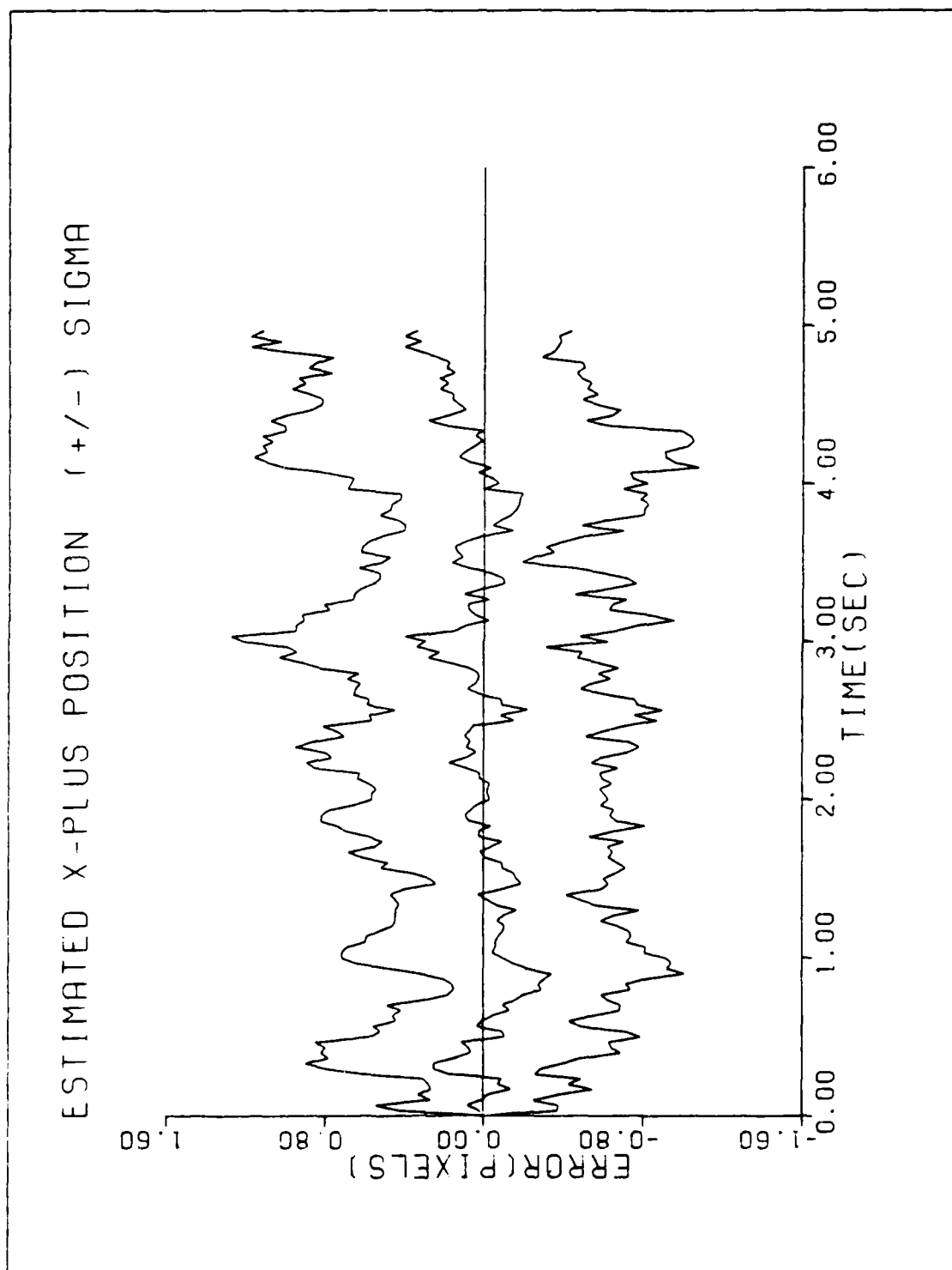


Figure D-5. GM/SNG-S/T2/10-G/QB1

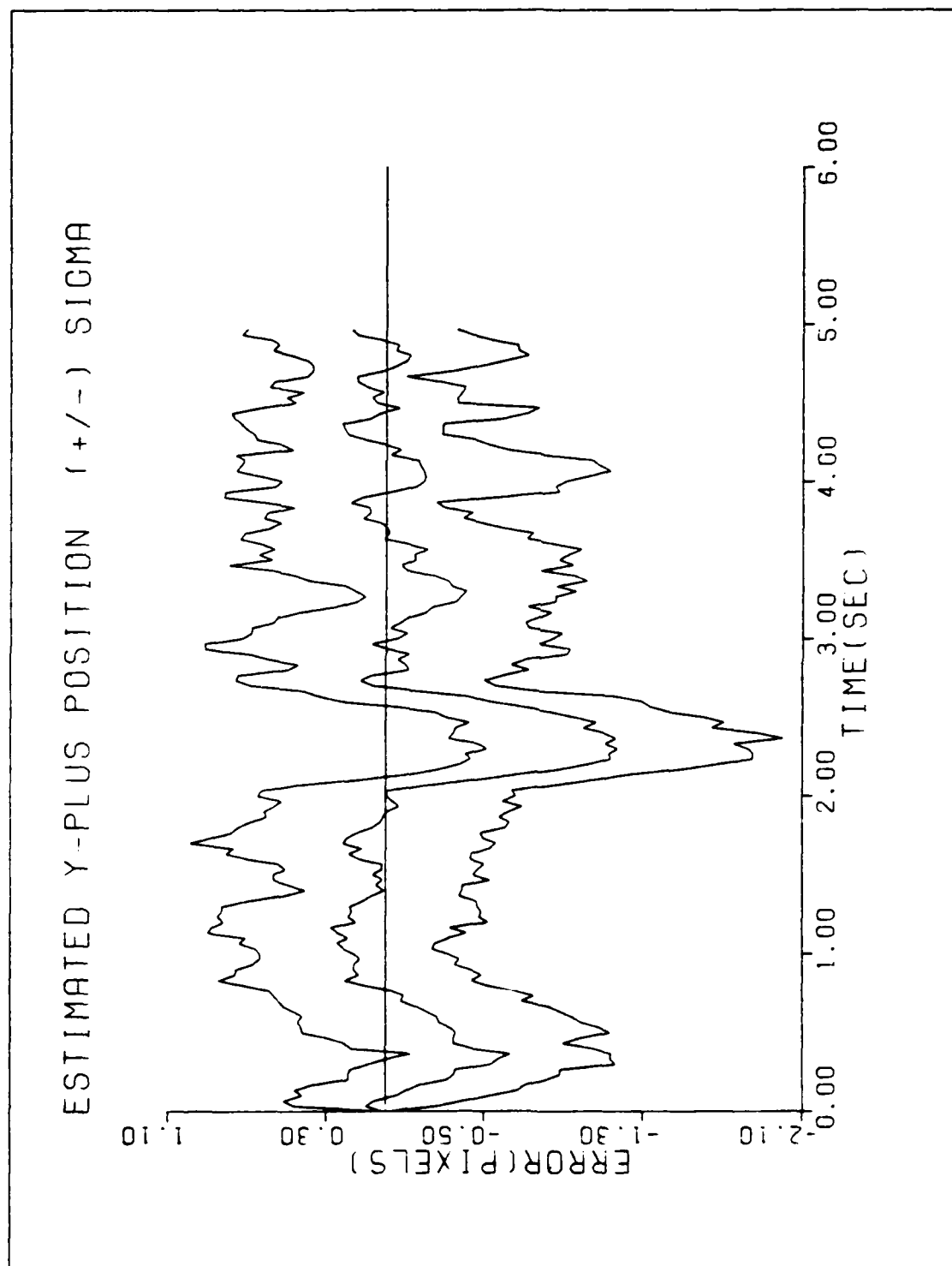


Figure D-6. GM/SNG-S/T2/10-G/QB1

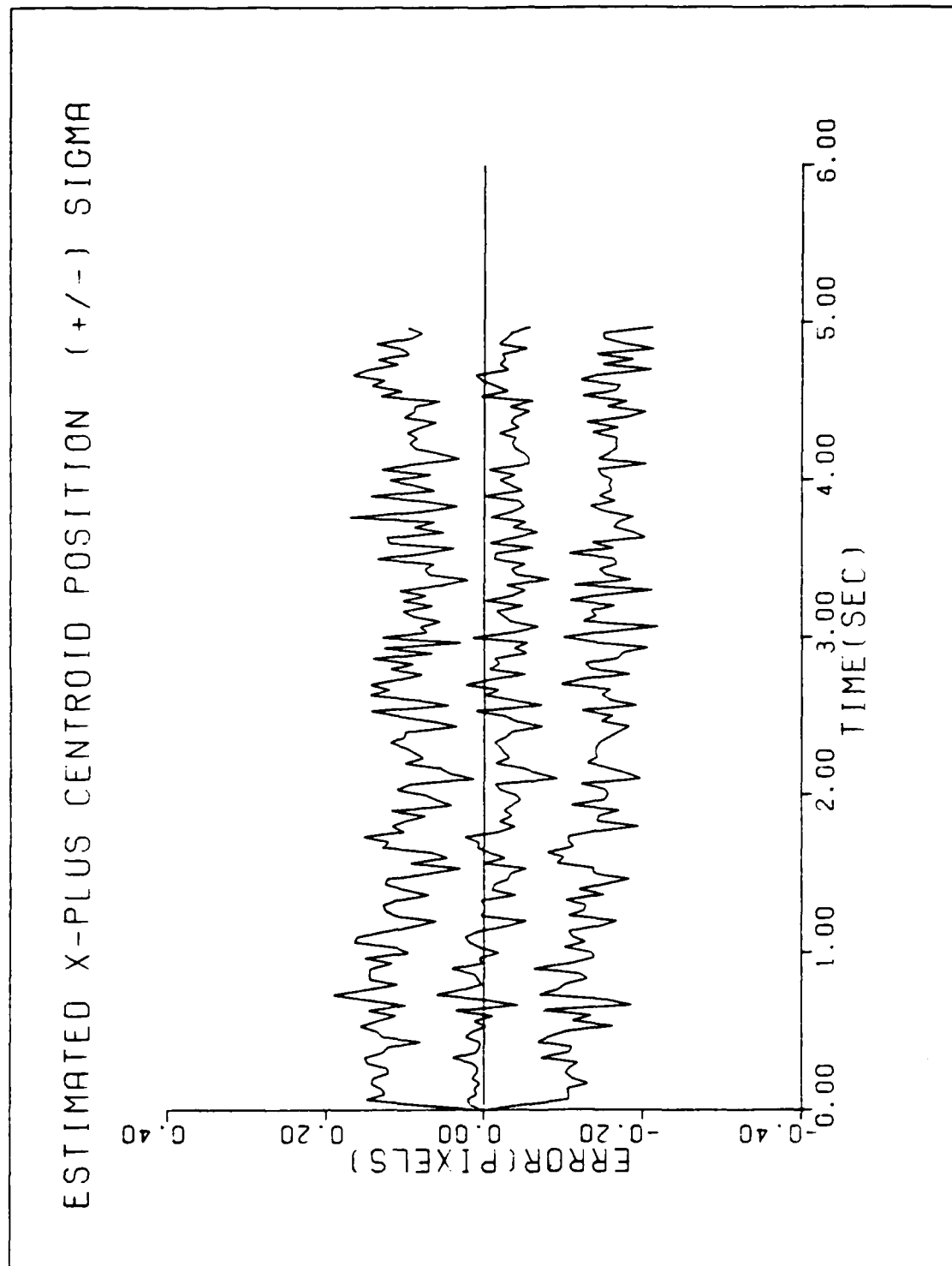


Figure D-7. GM/SNG-S/T2/10-G/QB1

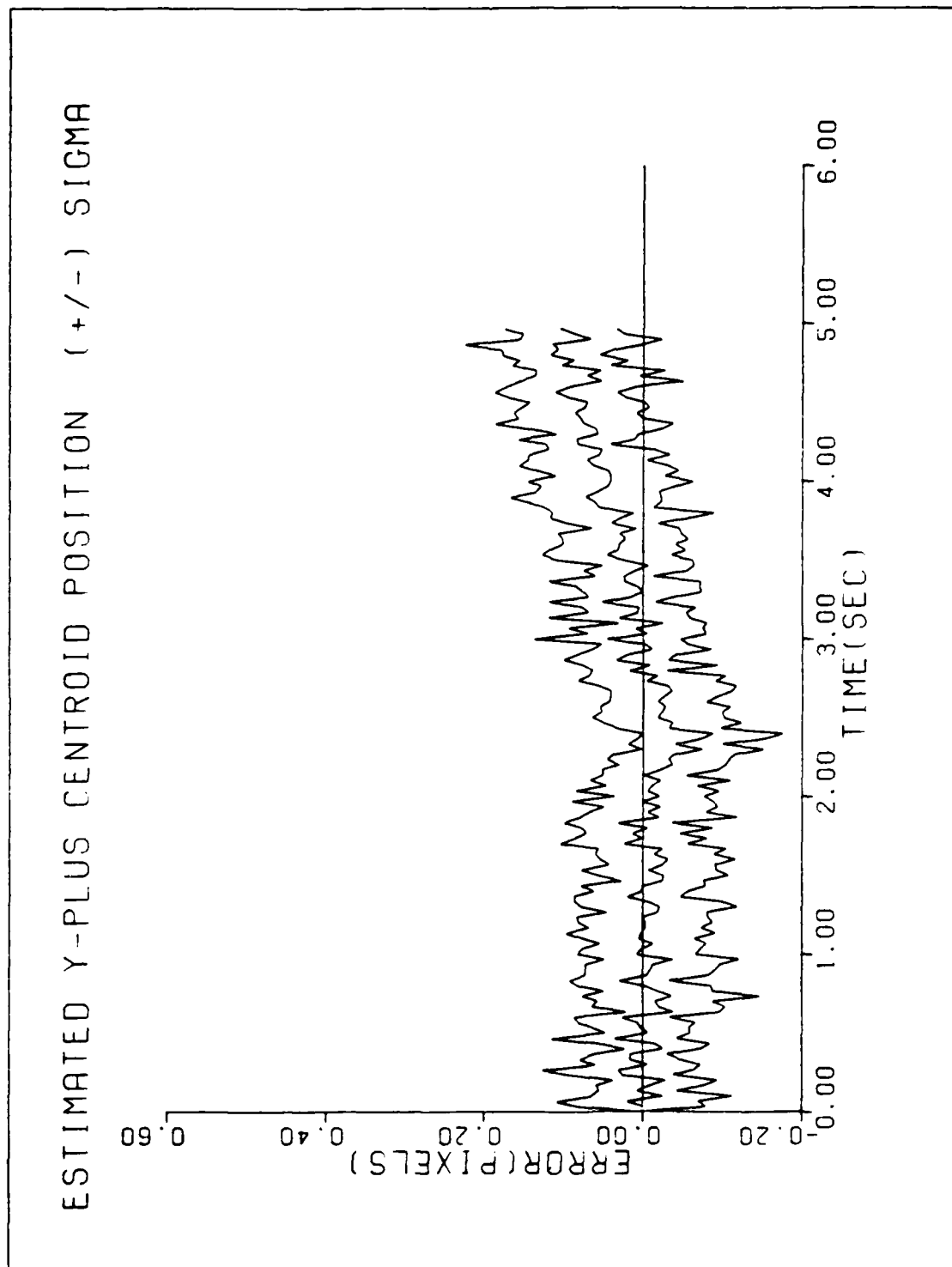


Figure D-8. GM/SNG-S/T2/10-6/QB1

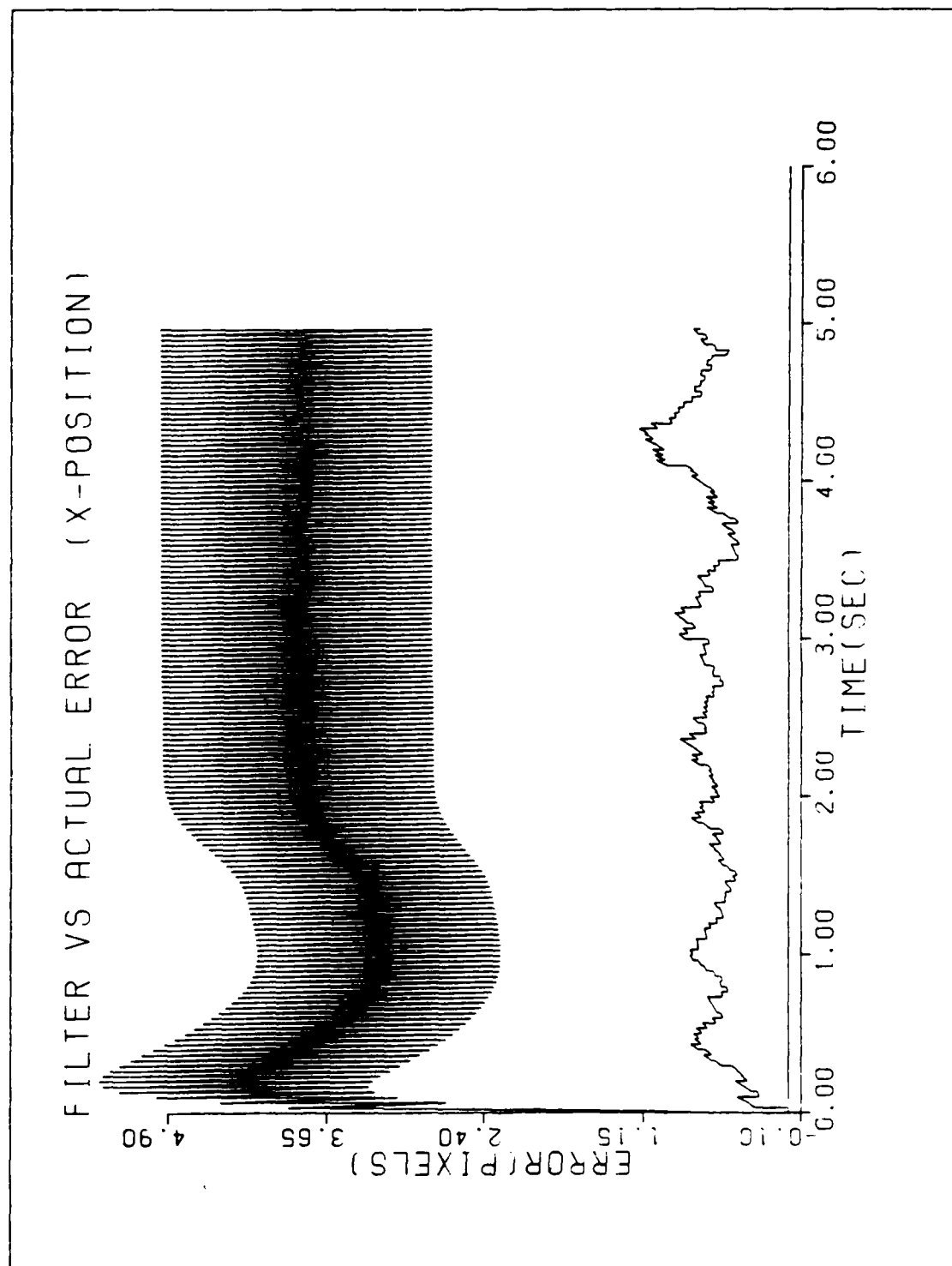


Figure D-9. CTR/SNG-S/T2/10-G/QB1

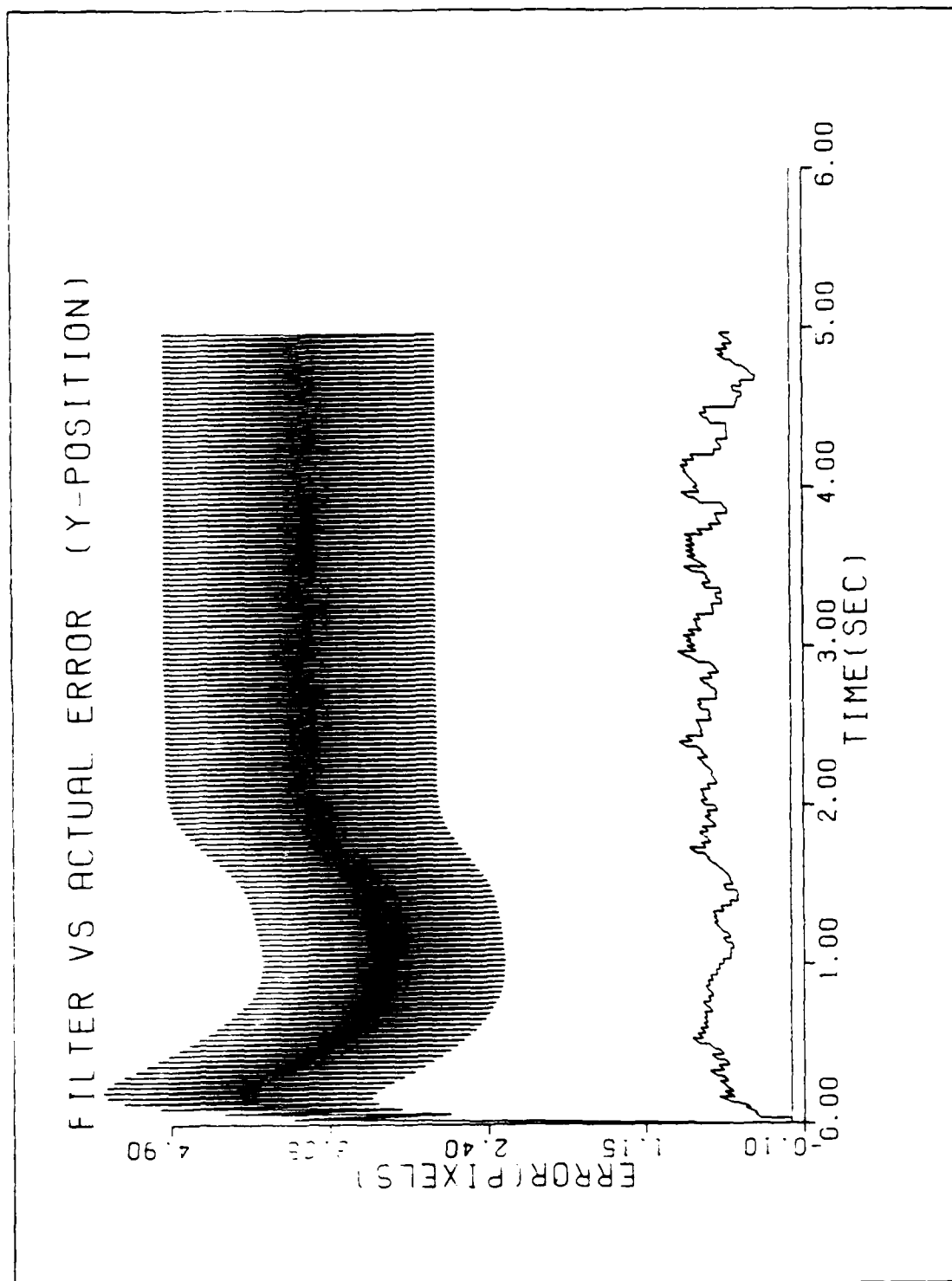


Figure D-10. CTR/SNG-S/T2/10-G/QB1

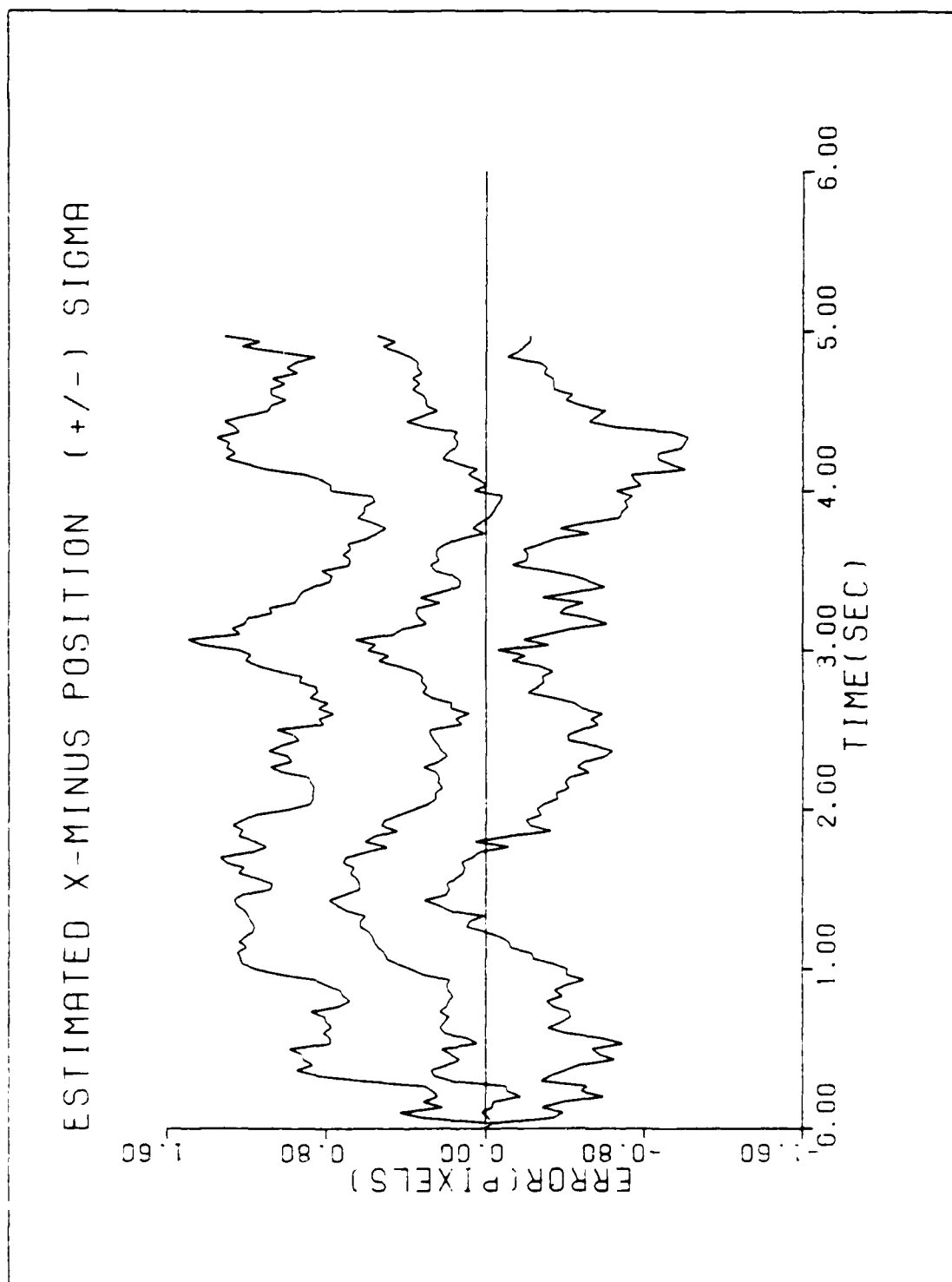


Figure D-11. CTR/SNG-S/T2/10-G/QB1

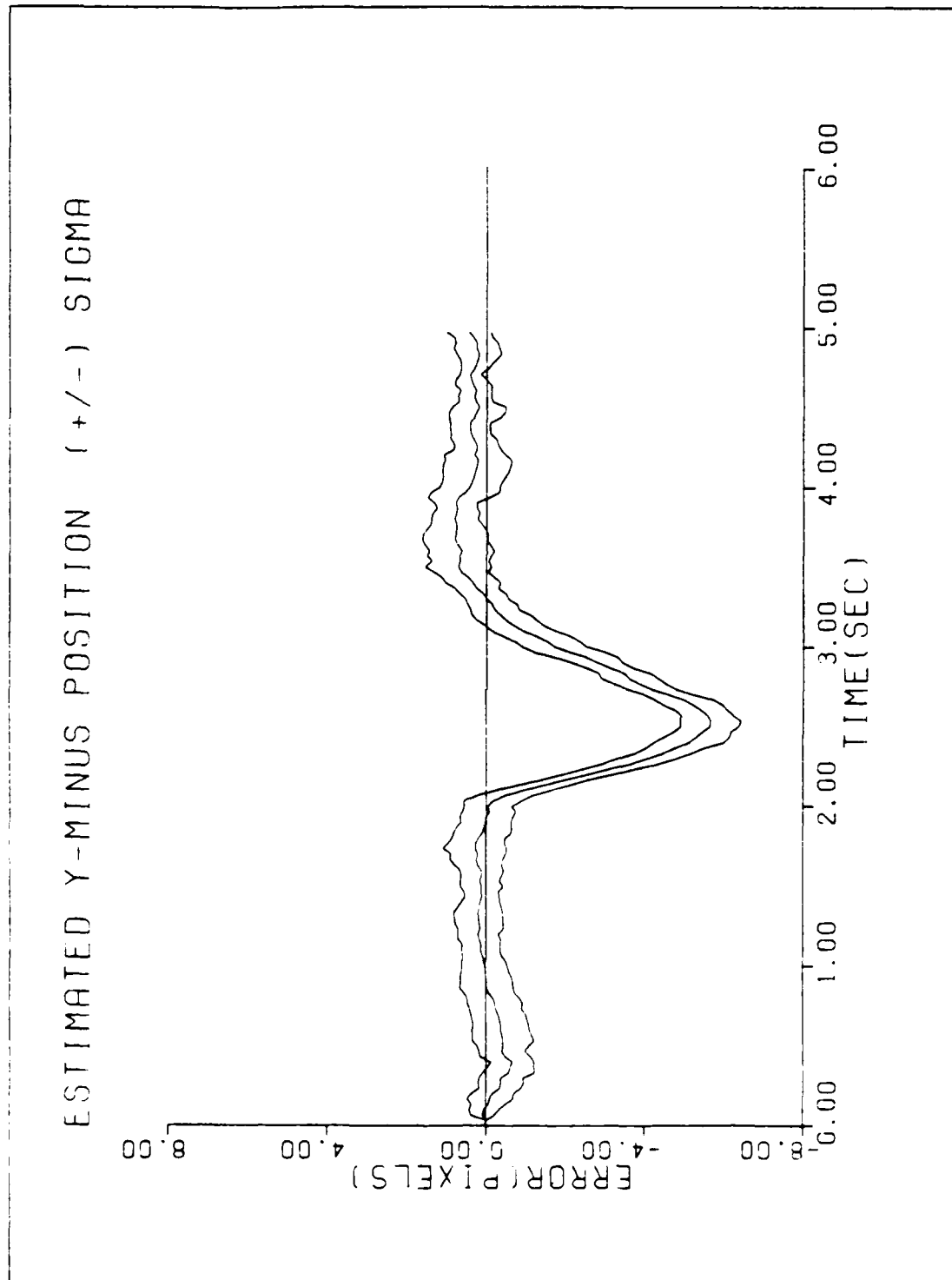


Figure D-12. CTR/SNG-S/T2/10-G/QB1

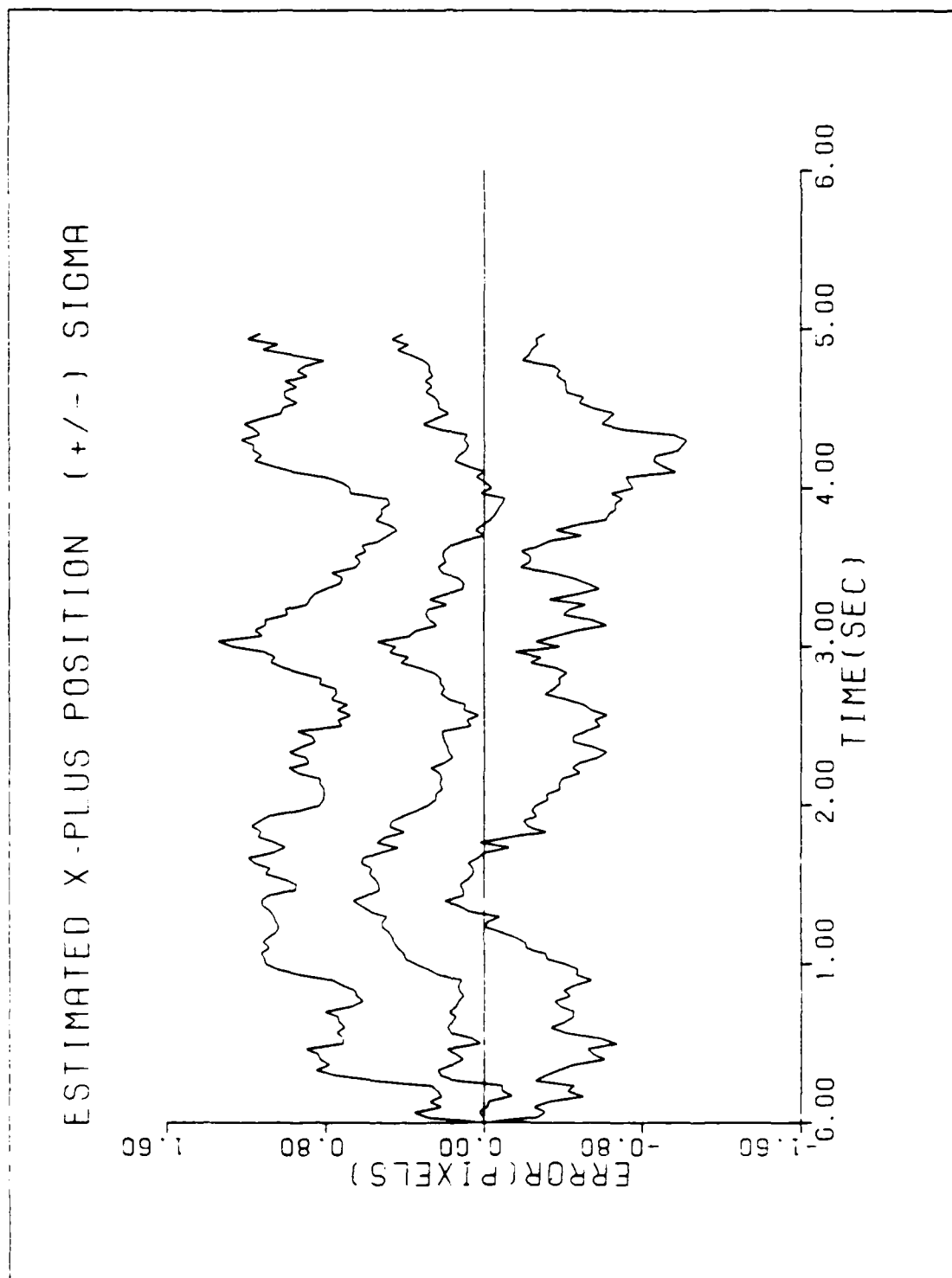


Figure D-13. CTR/SNG-S/T2/10-G/QB1

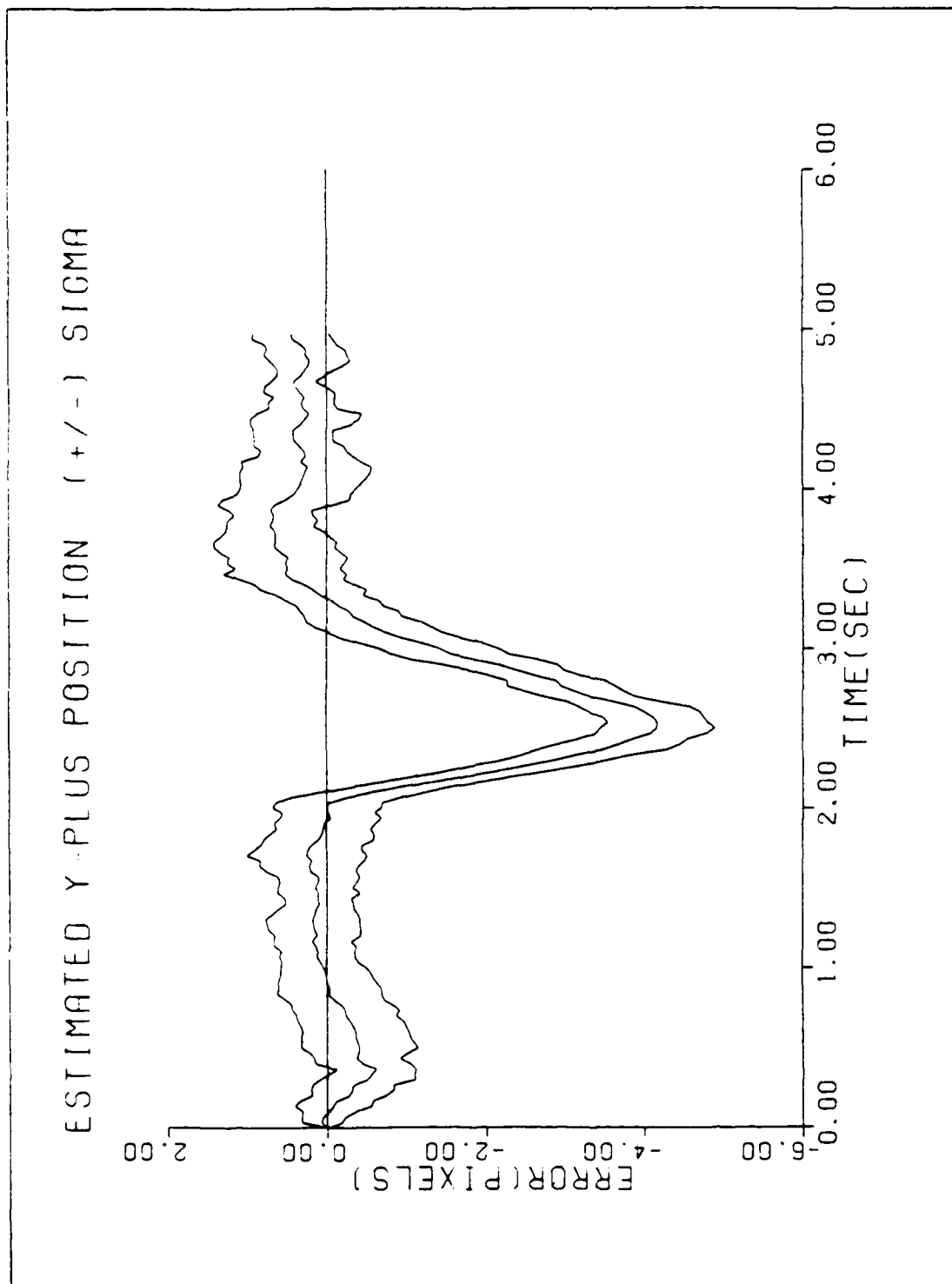


Figure D-14. CTR/SNG-S/T2/10-G/QB1

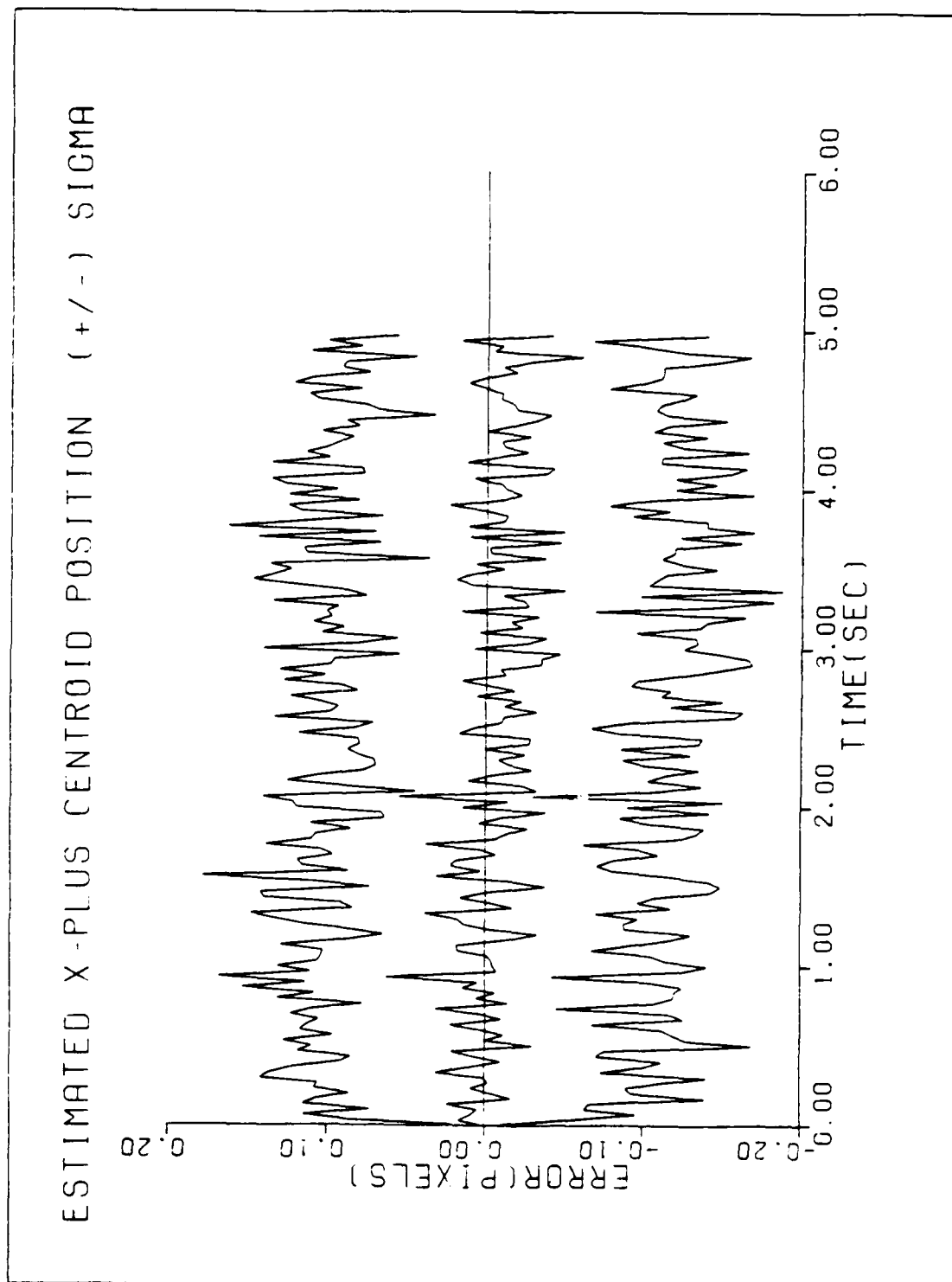


Figure D-15. CTR/SNG-S/T2/10-G/QB1

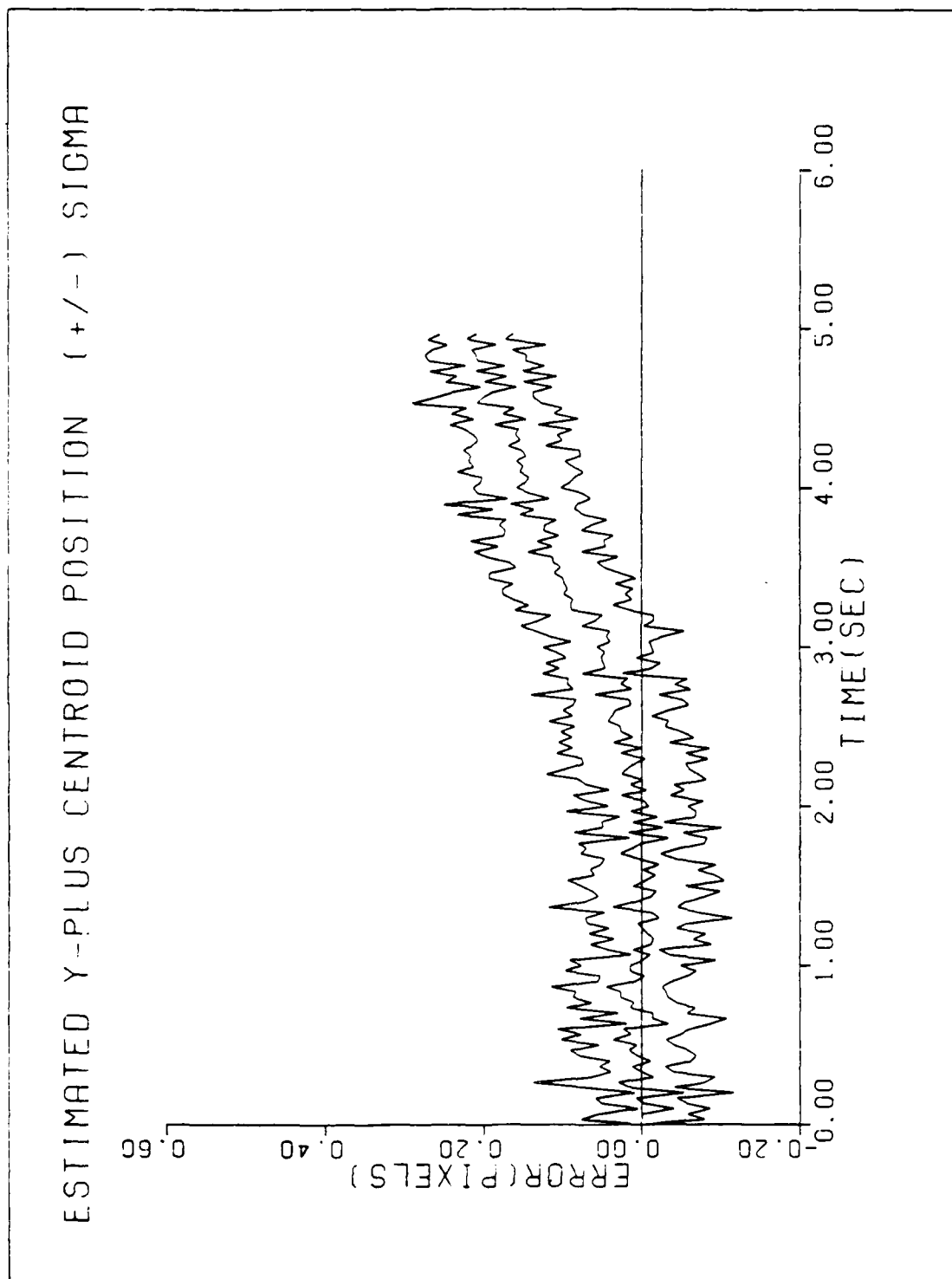


Figure D-16. CTR/SNG-S/T2/10-G/QB1

APPENDIX E

MULTIPLE MODEL ADAPTIVE FILTER SIMULATIONS BENCHMARK AND BENDING/VIBRATION ANALYSIS (REFERENCED FROM SECTION 6.4)

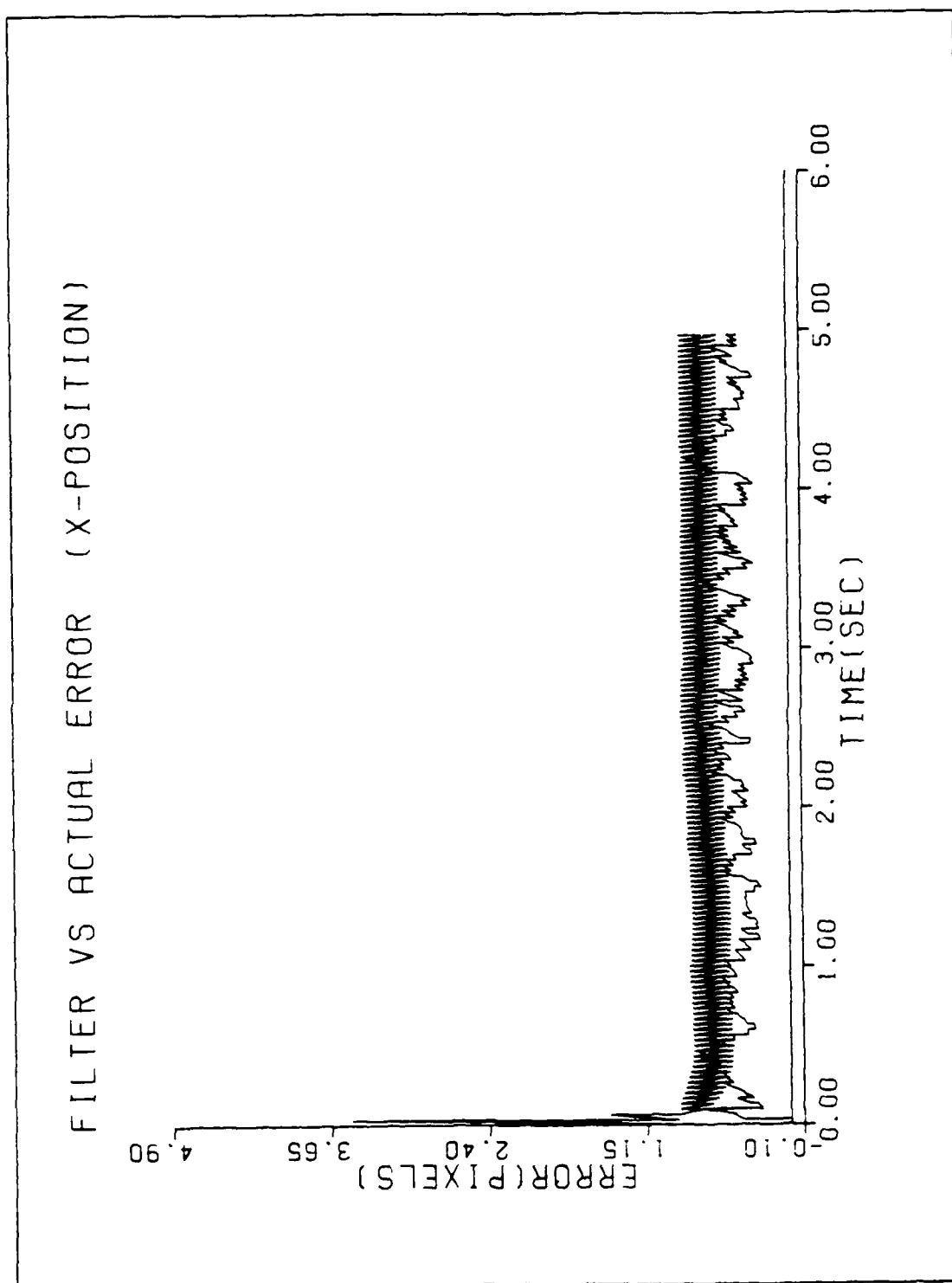


Figure E-1. GM/MMAF/T2/10-G

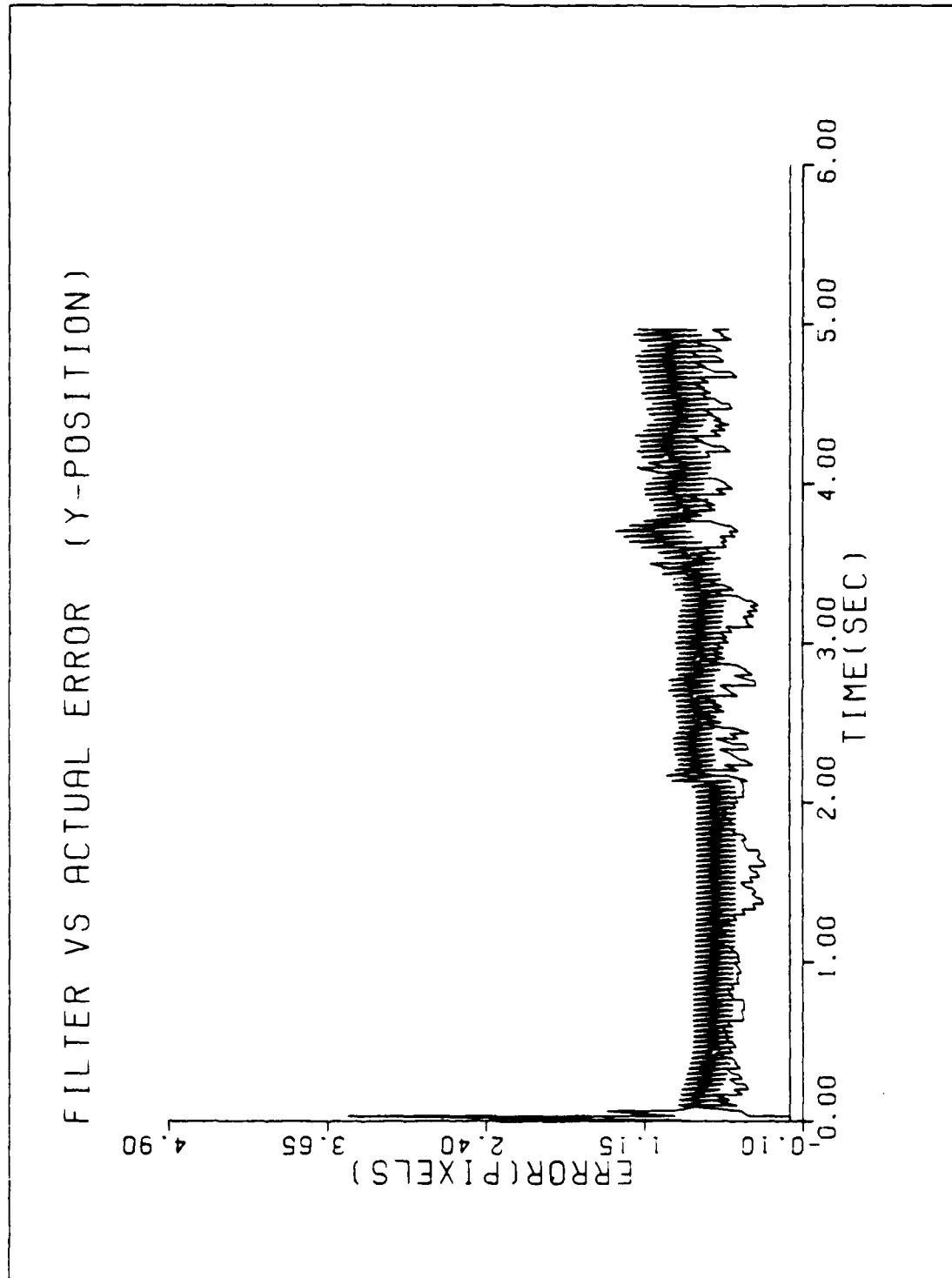


Figure E-2. GM/MMAF/T2/10-6

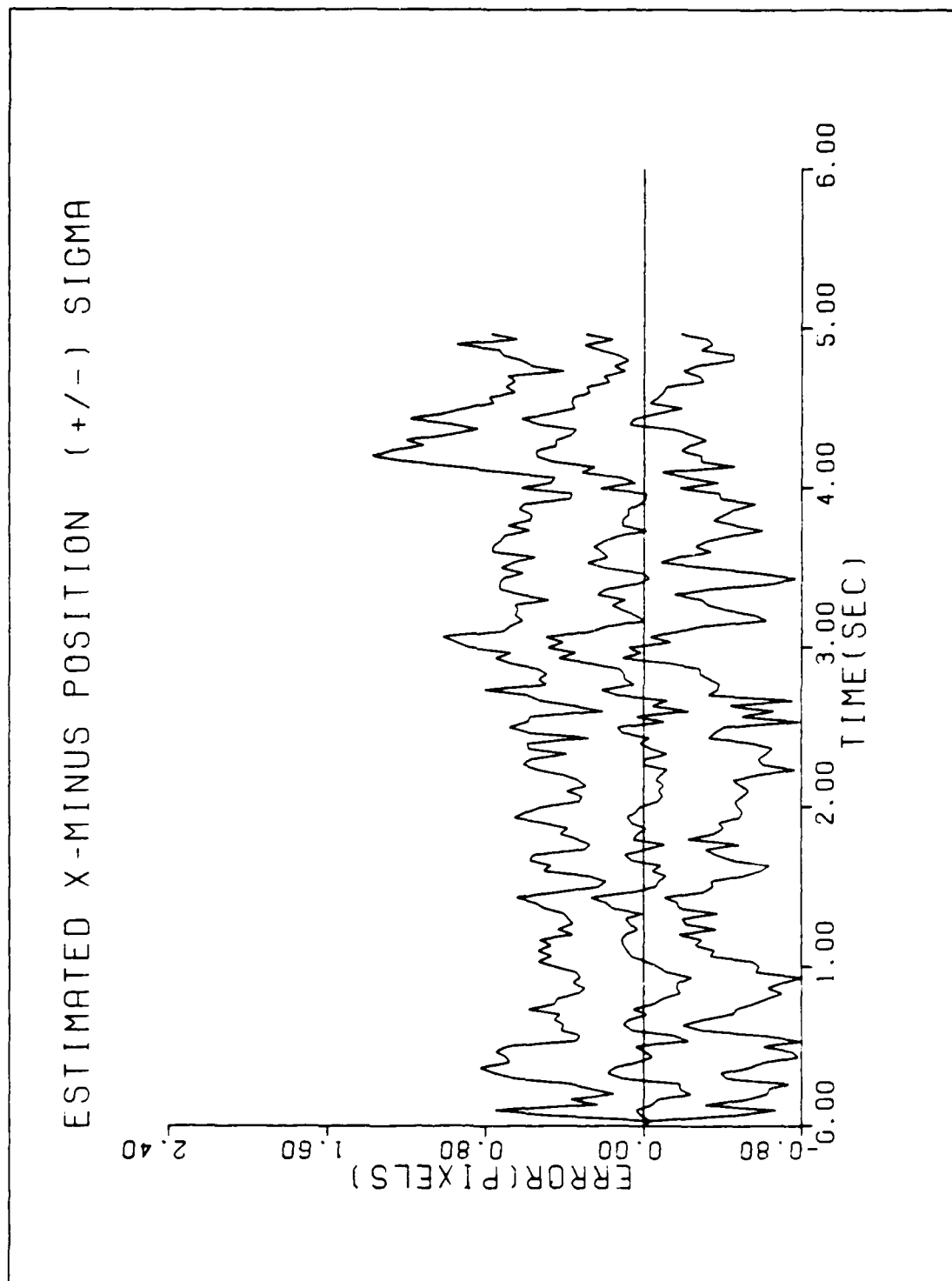


Figure E-3. GM/MMAF/T2/10-6

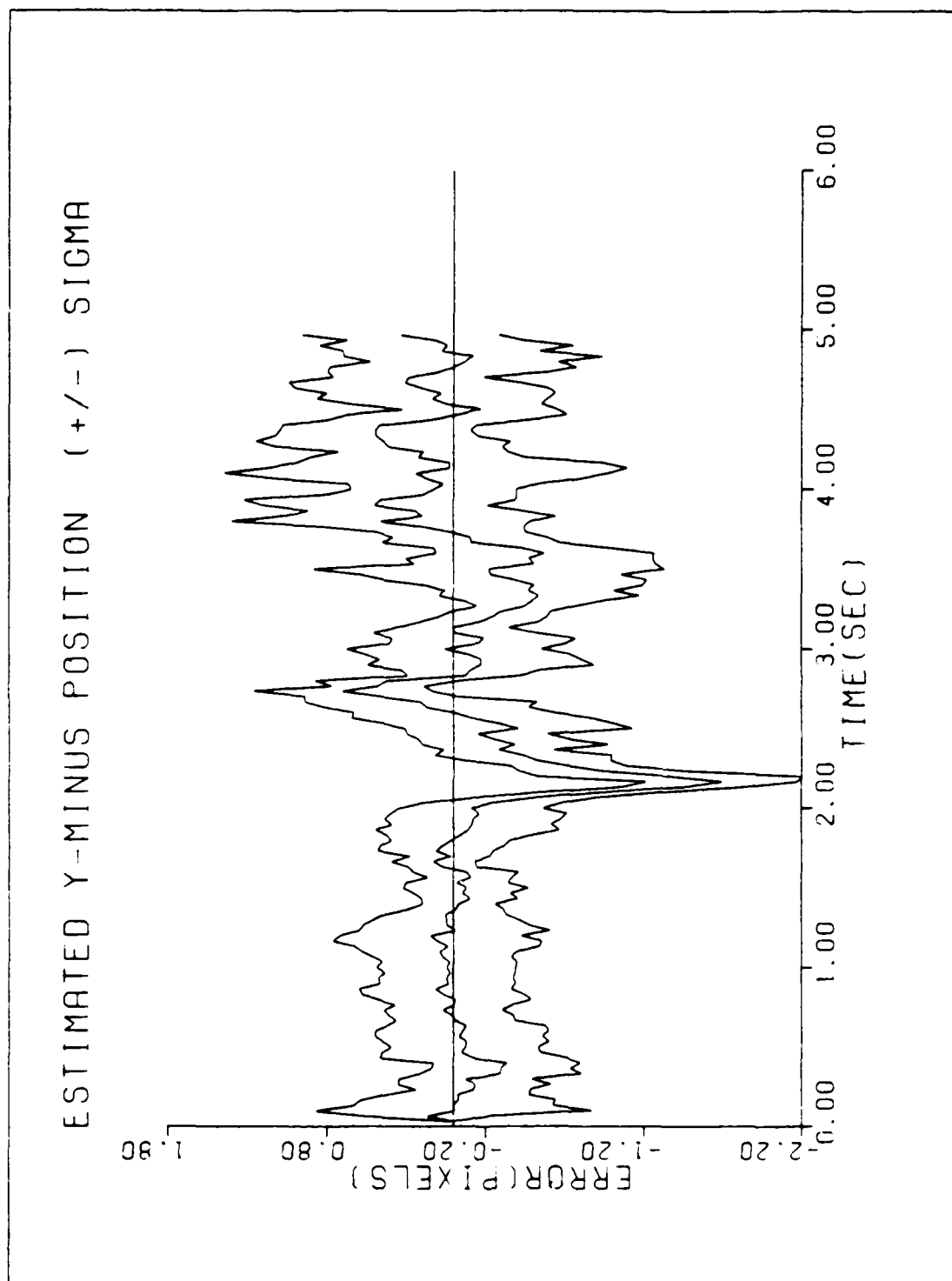


Figure E-4. GM/MMAF/T2/10-6

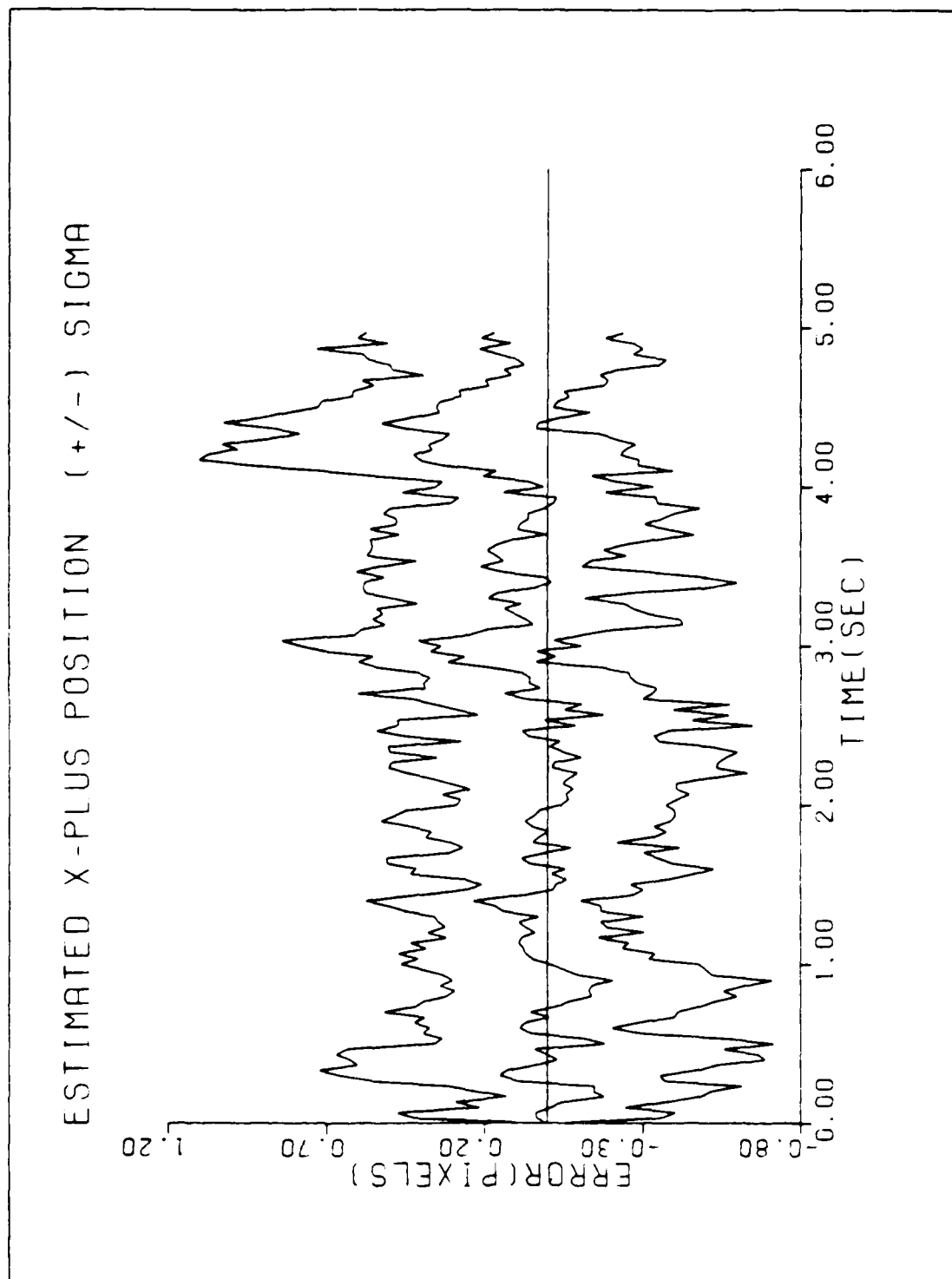


Figure E-5. GM/MMAF/T2/10-6

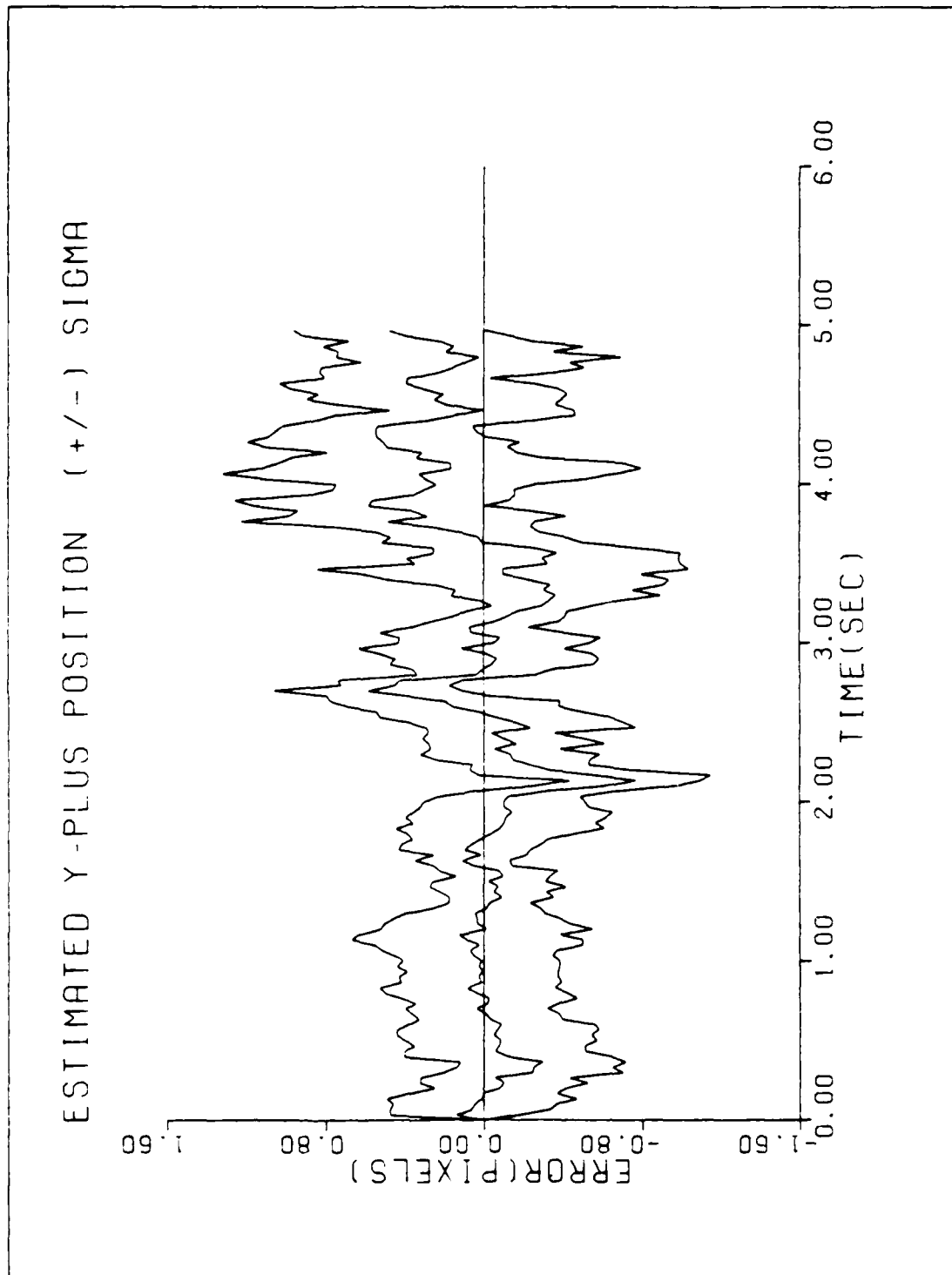


Figure E-6. GM/MMAF/T2/10-6

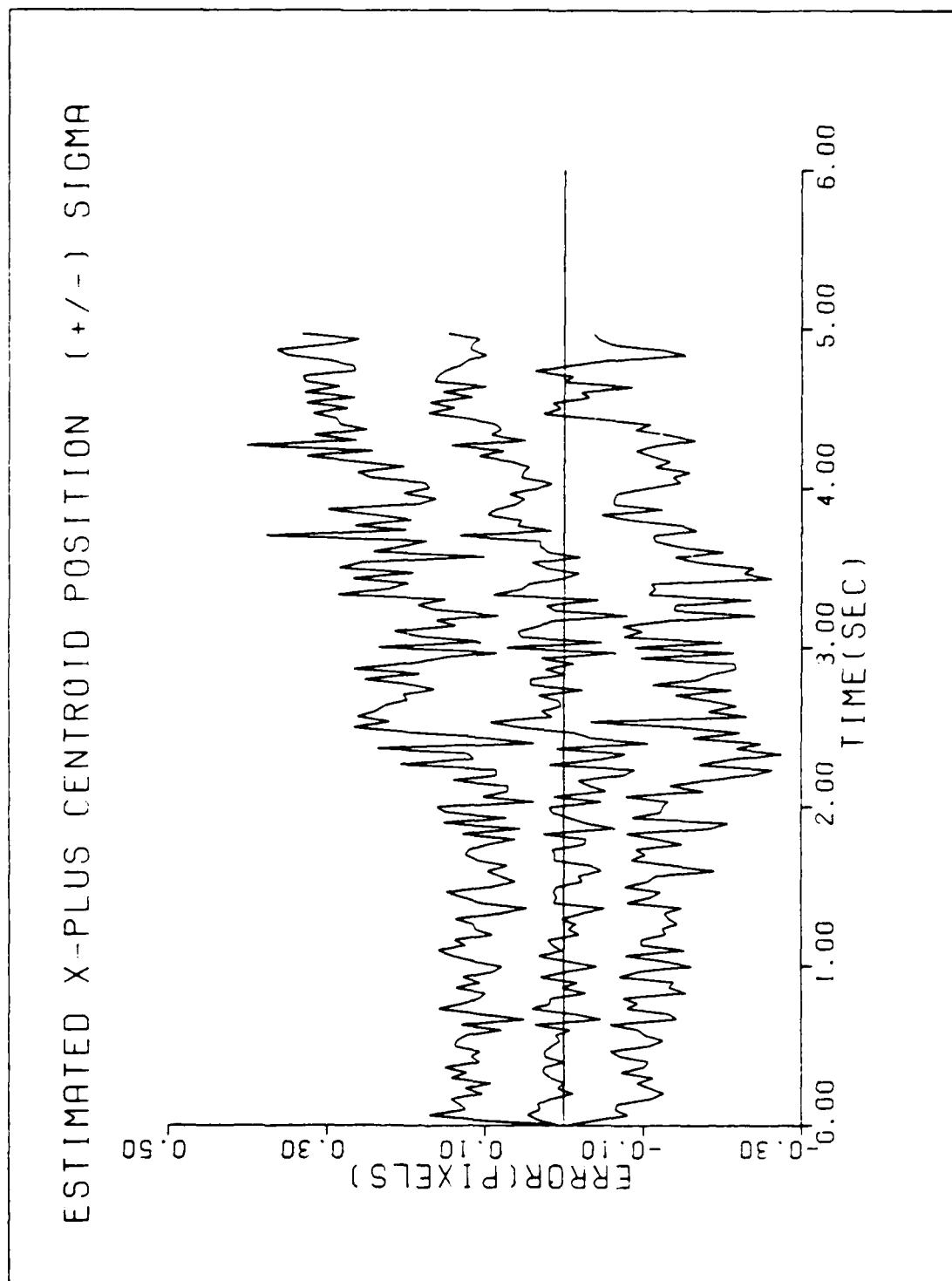


Figure E-7. GM/MMAF/T2/10-G

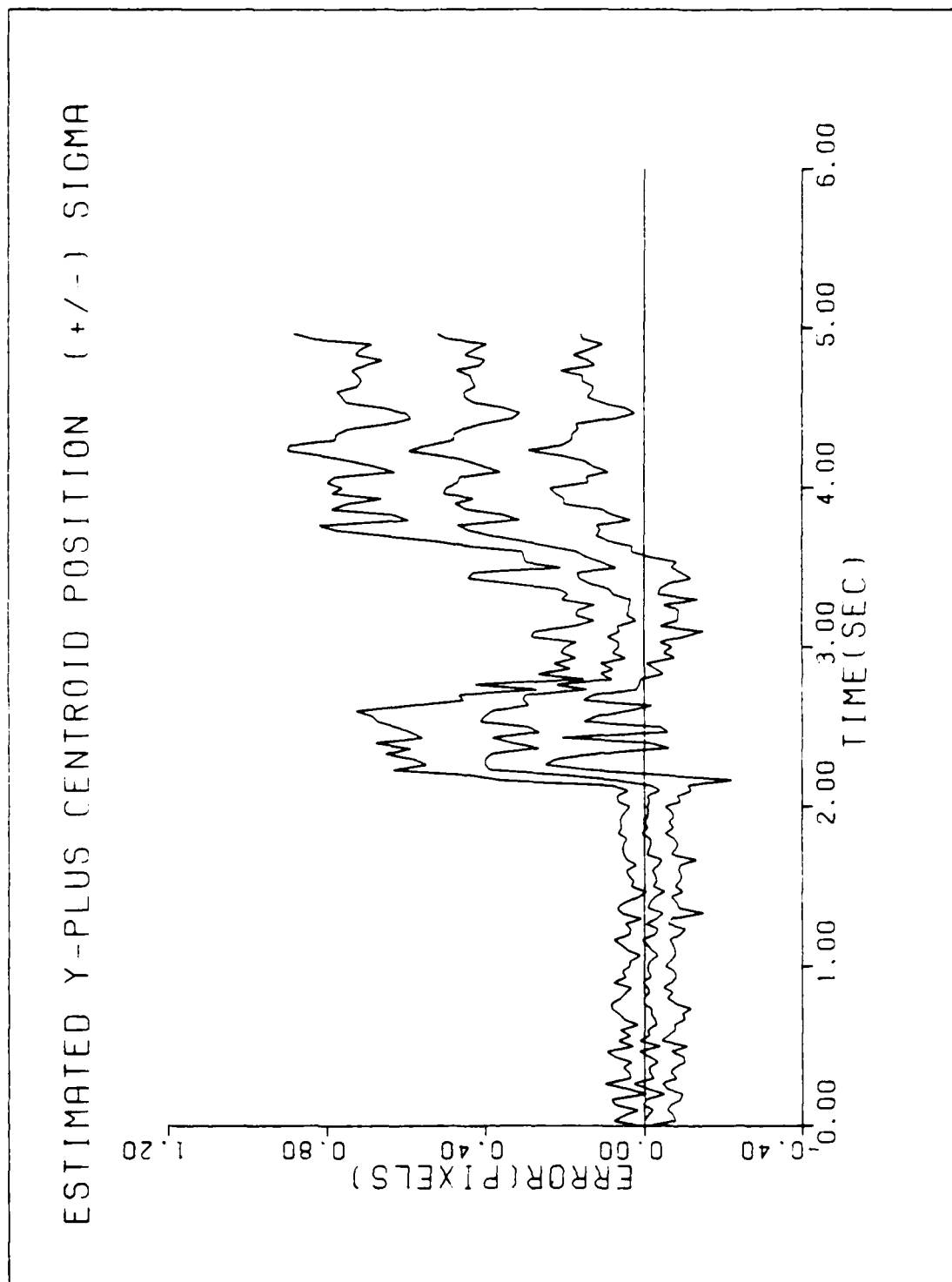


Figure E-8. GM/MMAF/T2/10-G

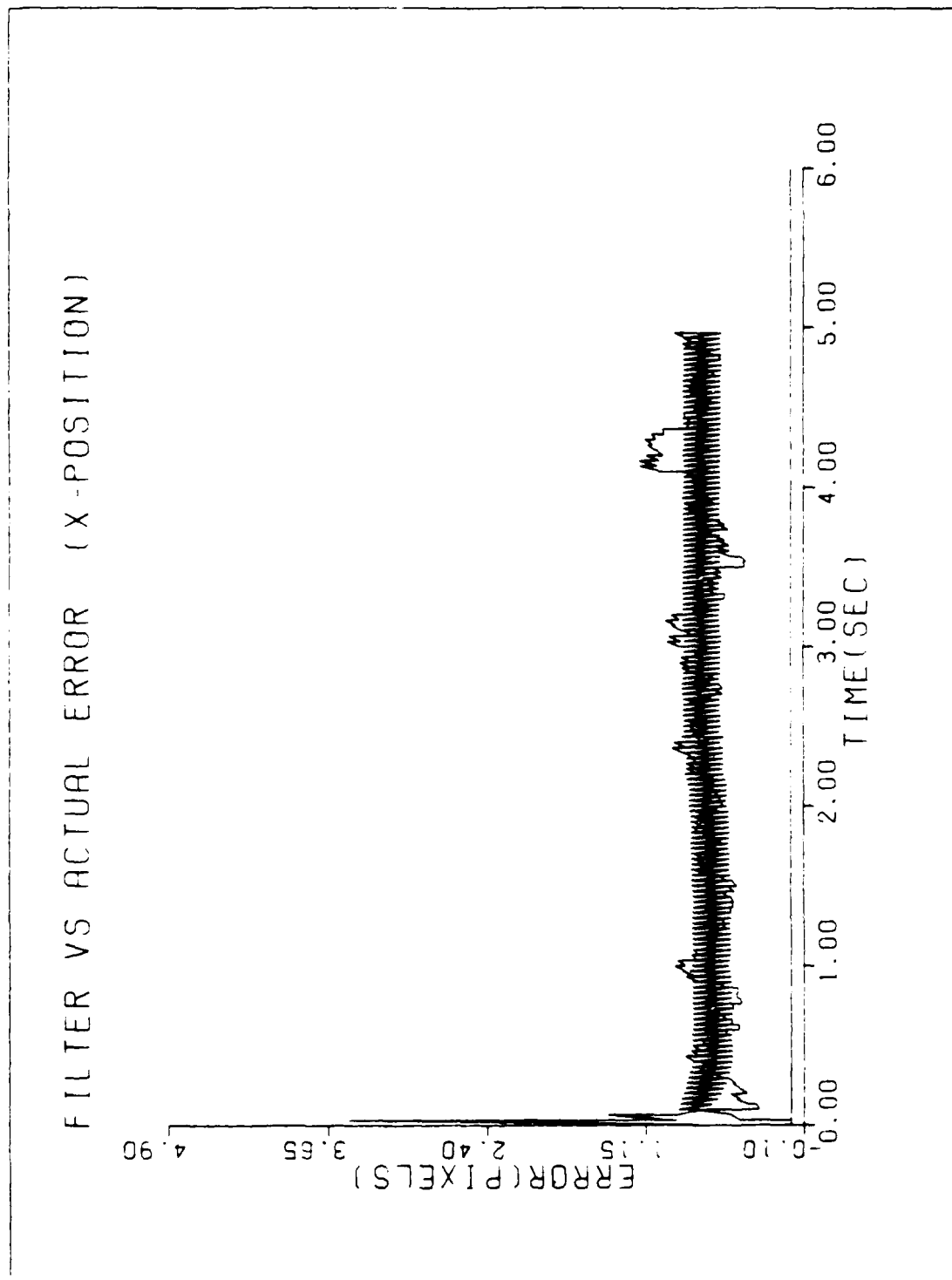


Figure E-9. GM/MMAF/T2/10-G/QB1

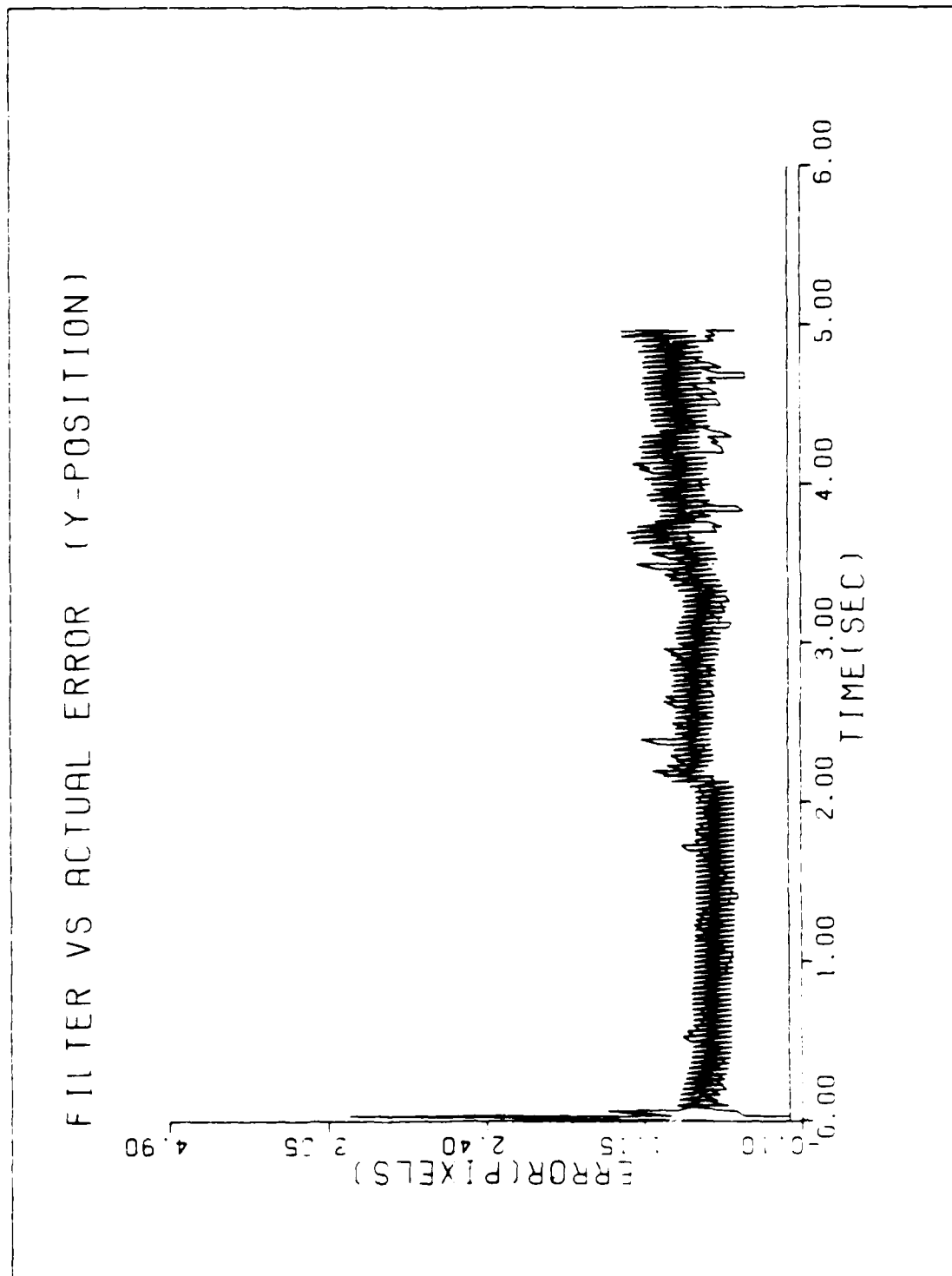


Figure E-10. GM/MMAF/T2/10-G/QB1

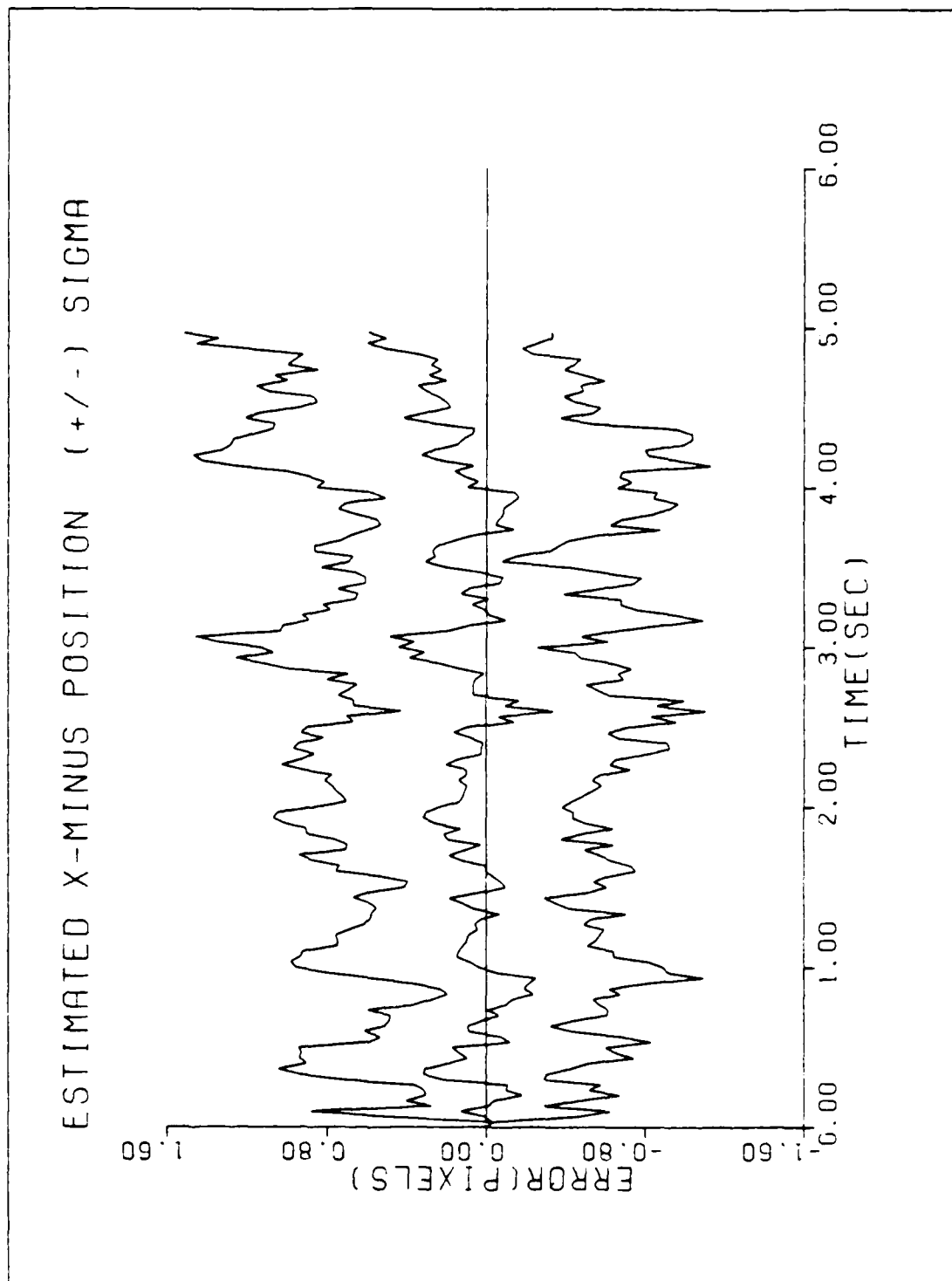


Figure E-11. GM/MMAF/T2/10-G/QB1

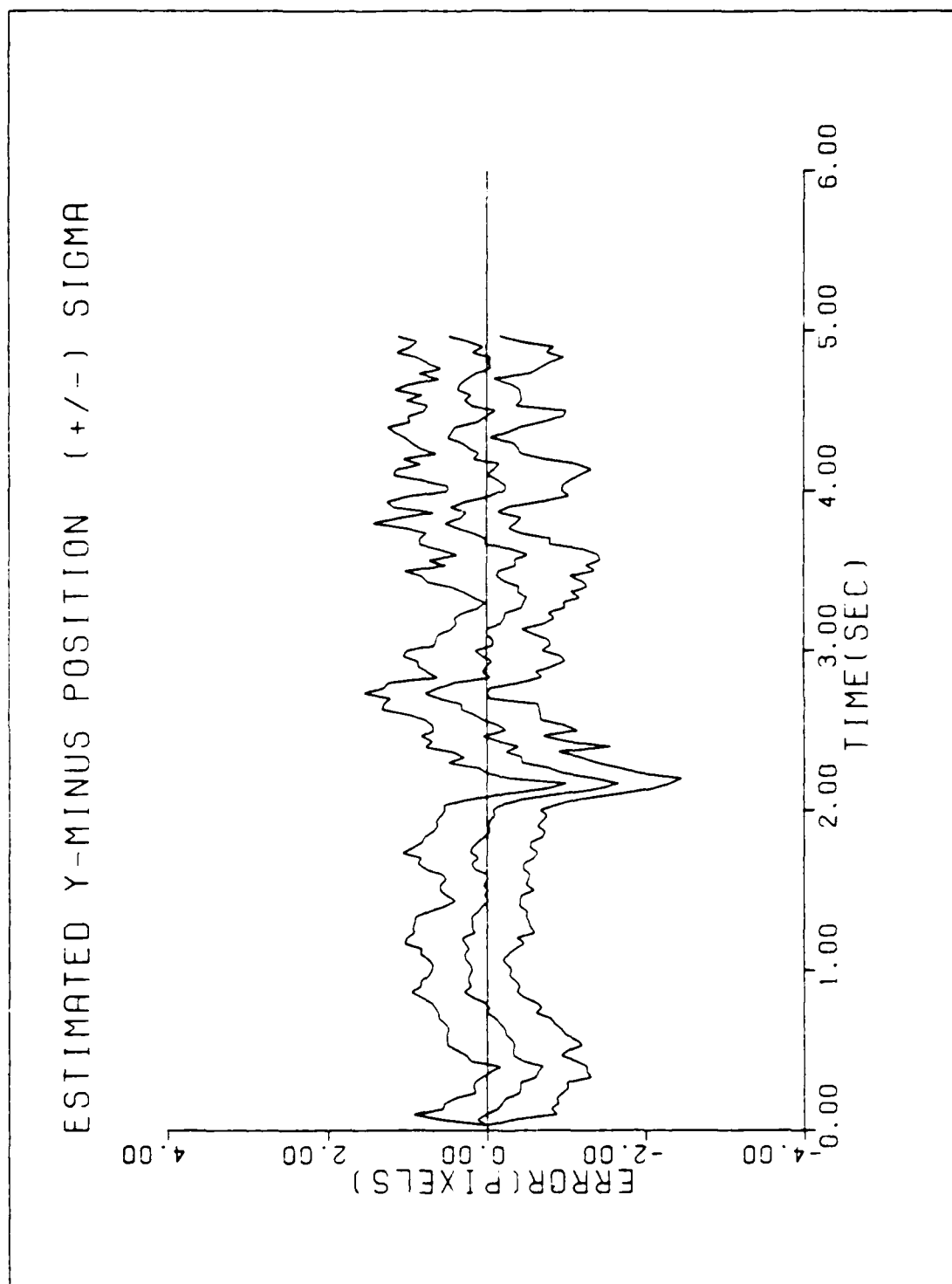


Figure E-12. GM/MAF/T2/10-G/QB1

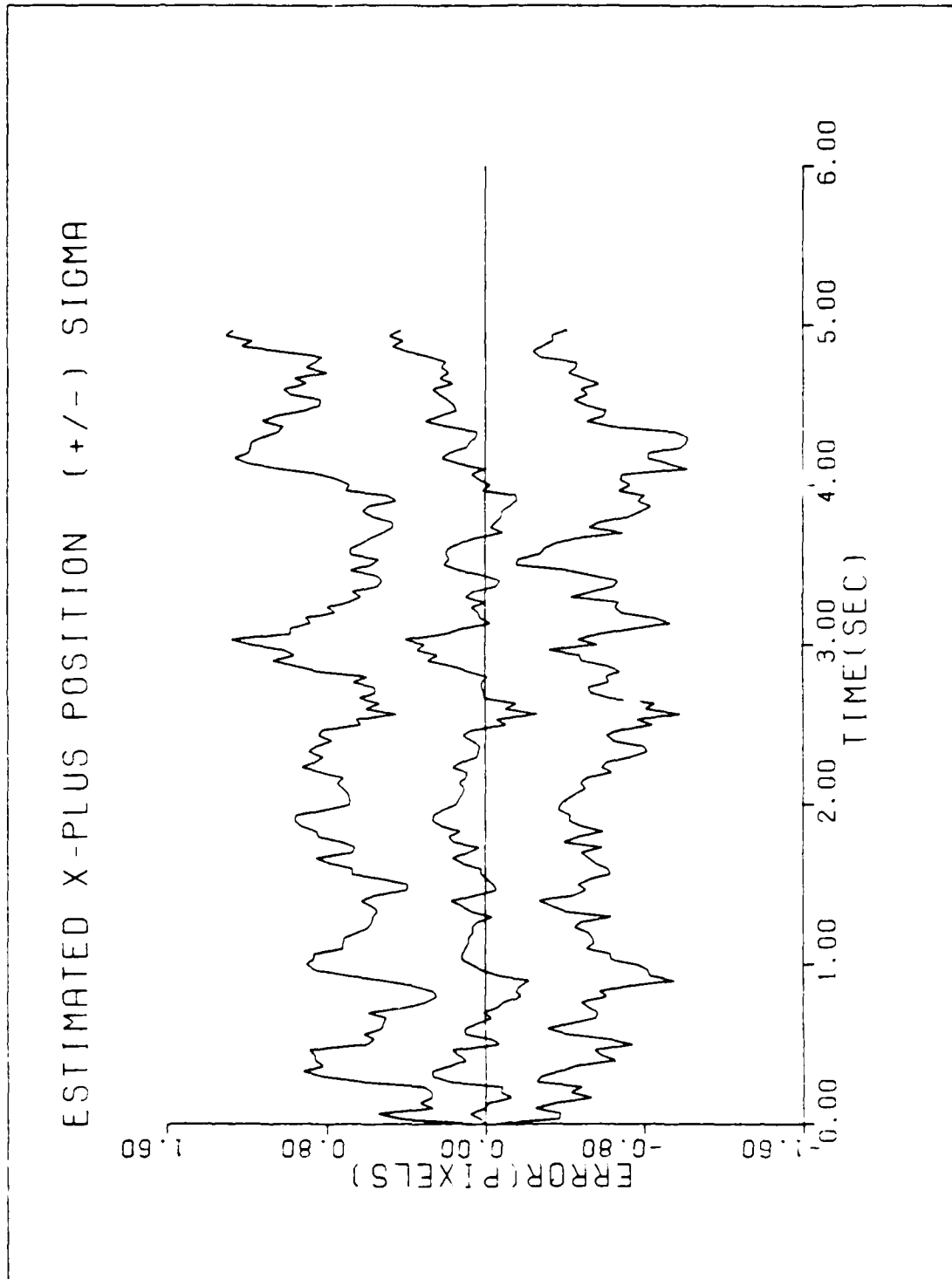


Figure E-13. GM/MMAF/T2/10-G/QB1

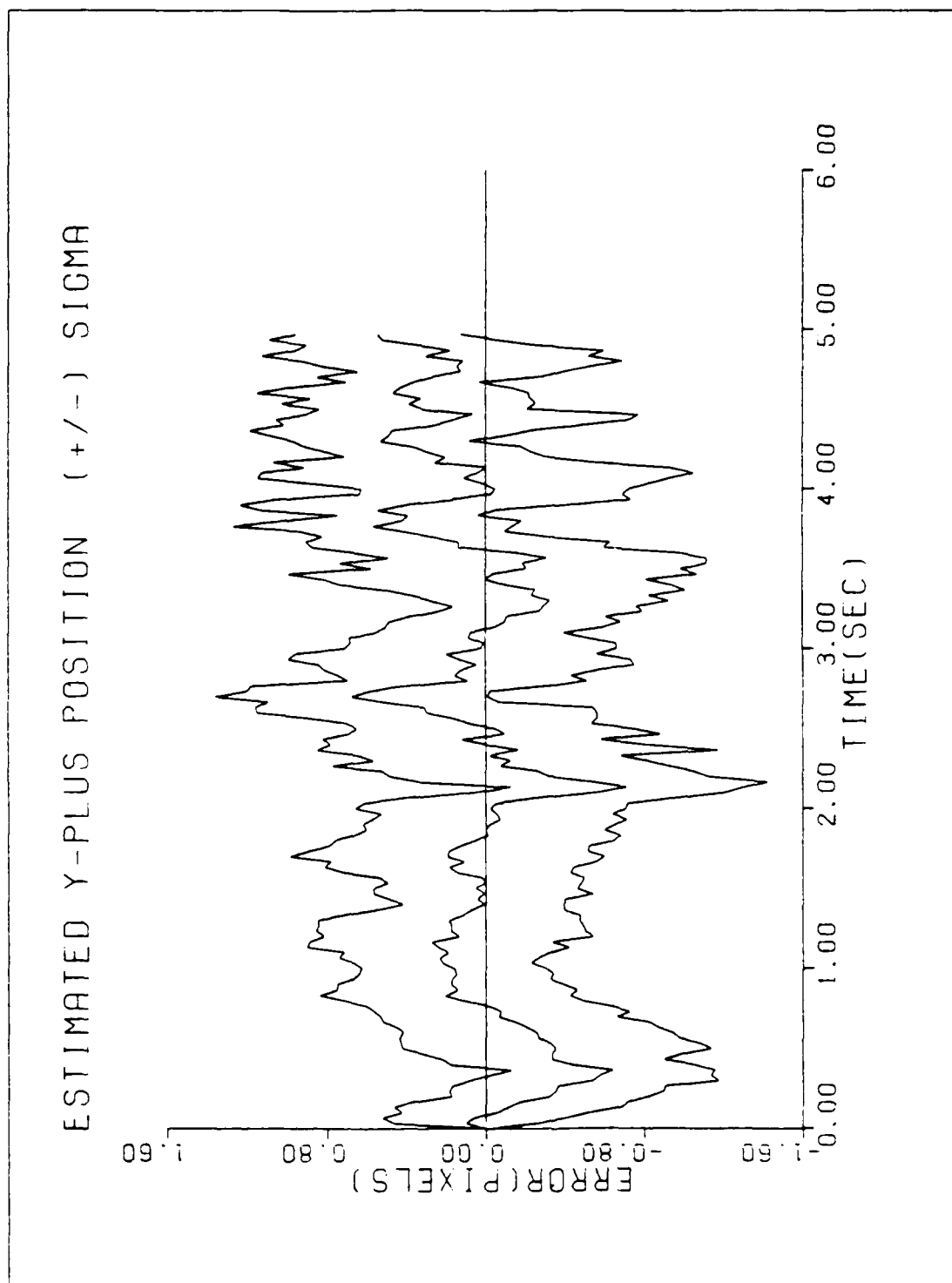


Figure E-14. GM/MMAF/T2/10-6/QB1

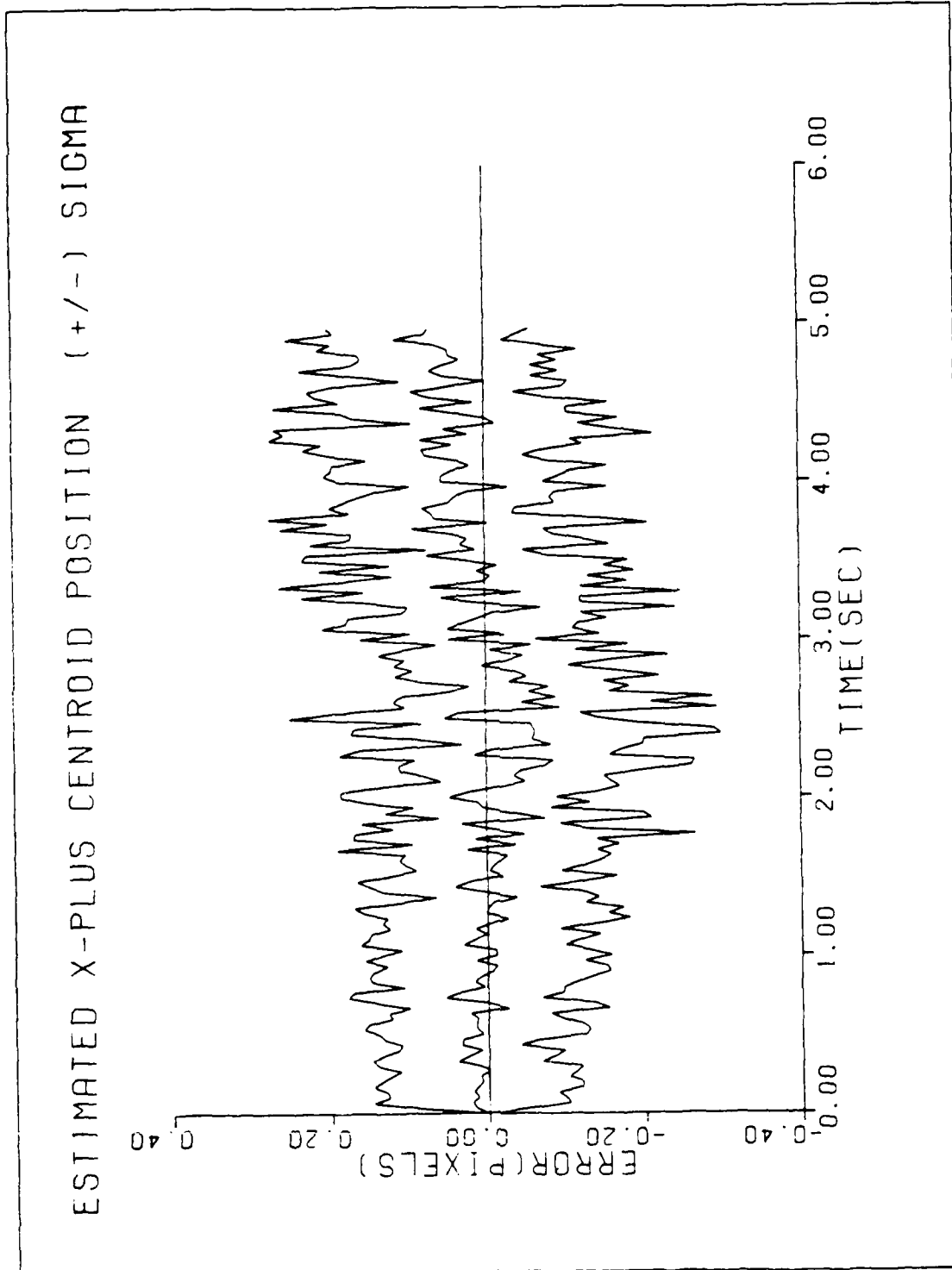


Figure E-15. GM/MMAF/T2/10-6/QB1

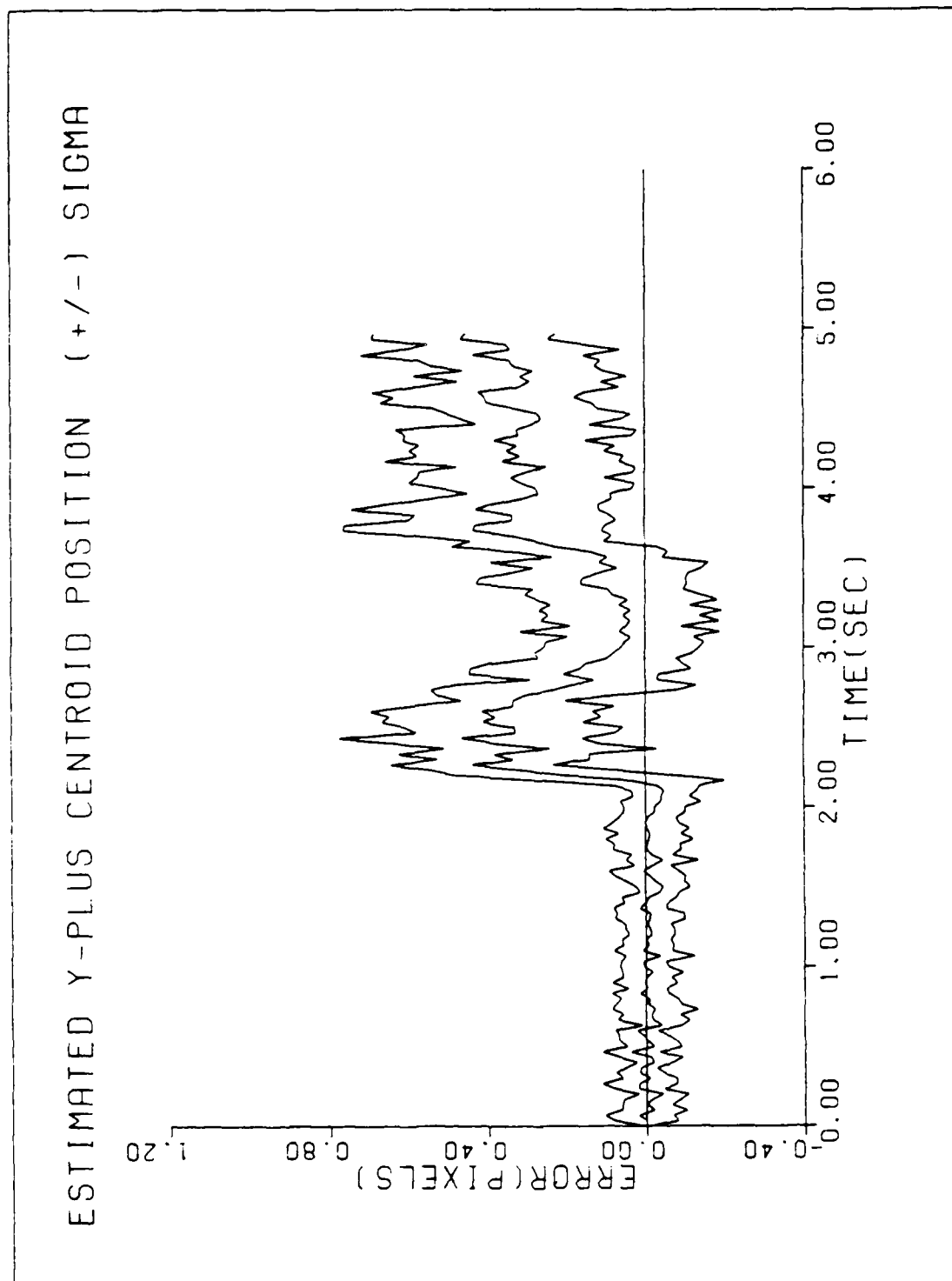


Figure E-16. GM/MMAF/T2/10-G/QB1

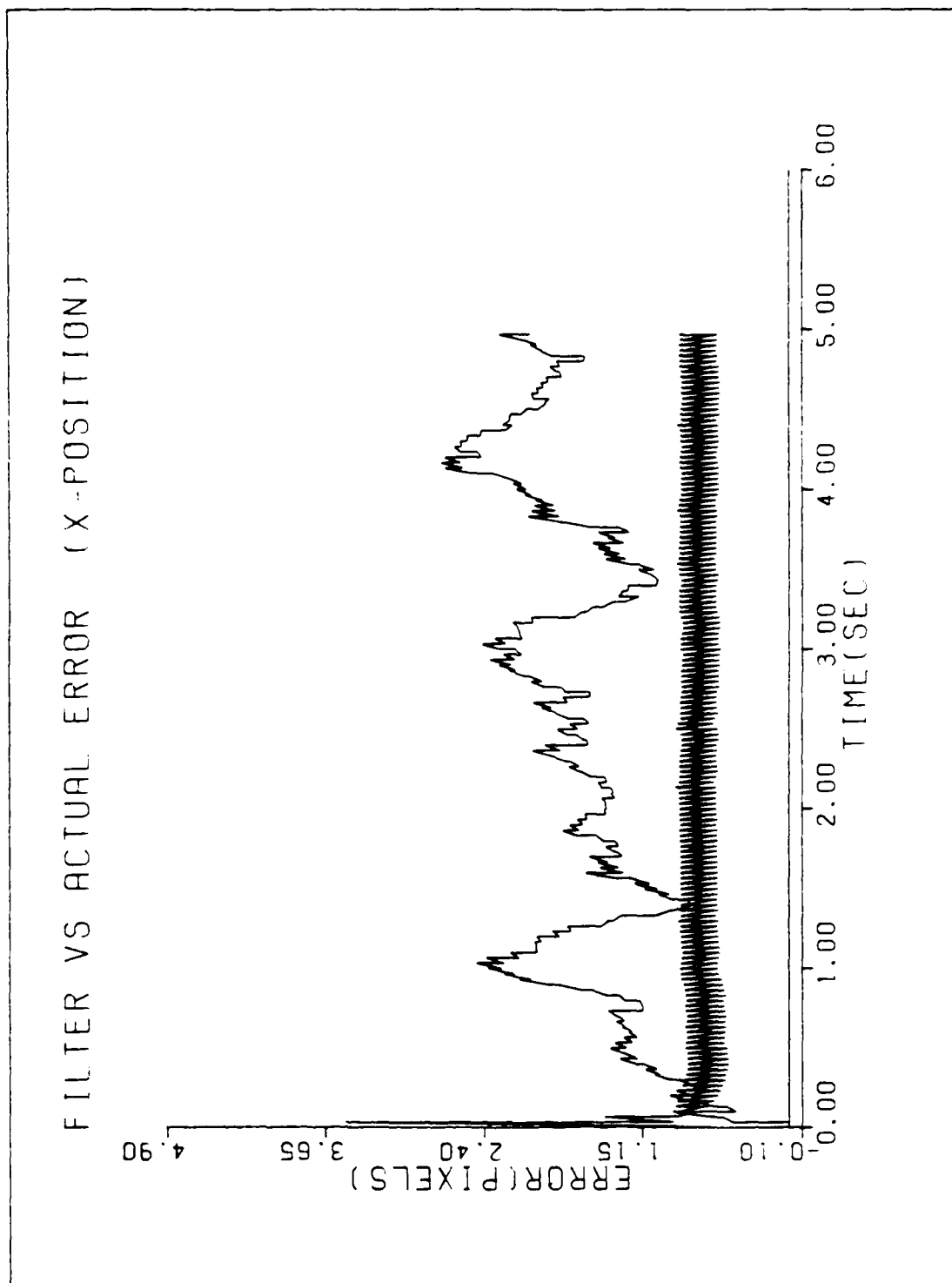


Figure E-17. GM/MMAF/T2/10-G/QB2

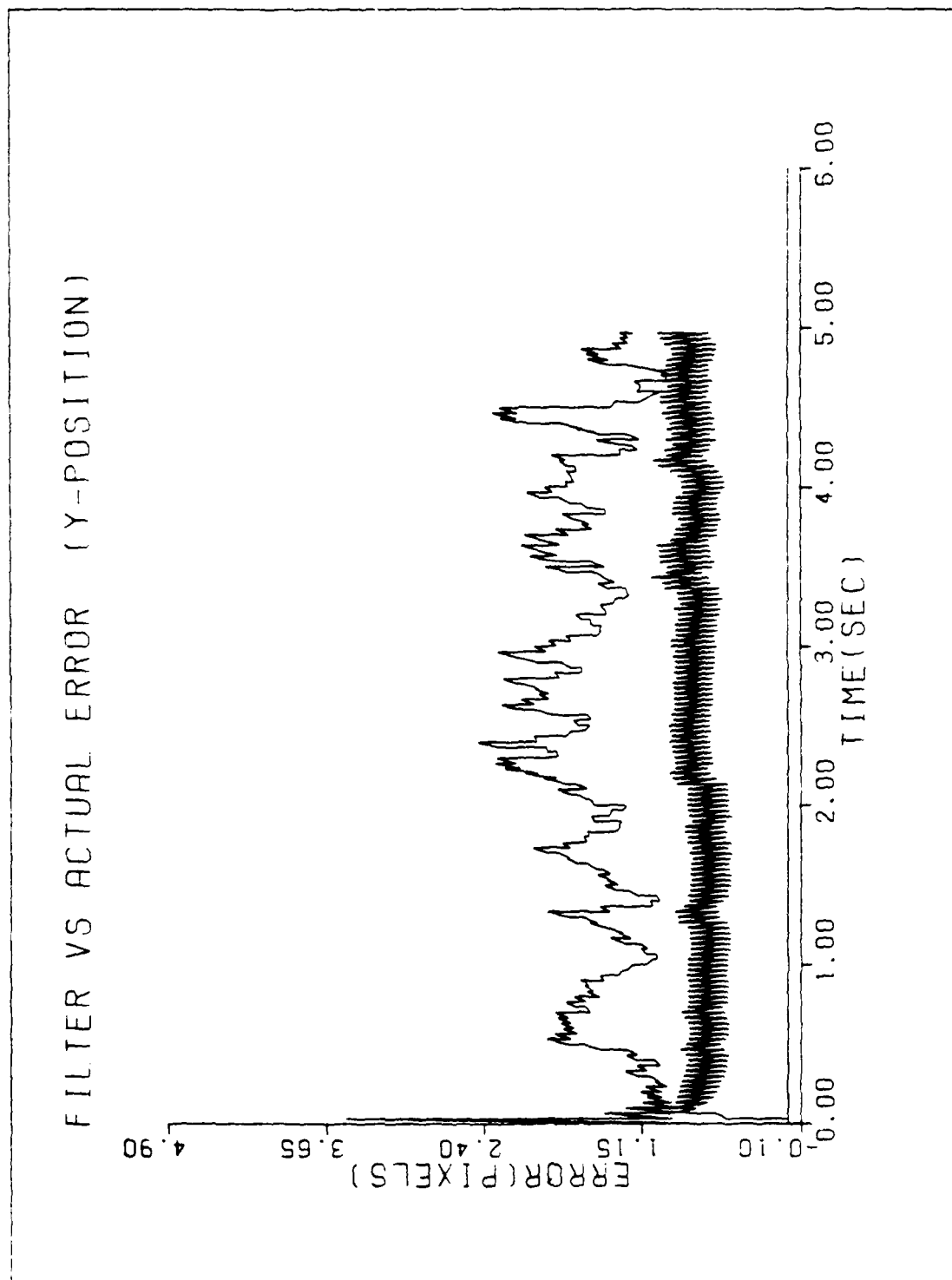


Figure E-18. GM/MMAF/T2/10-G/QB2

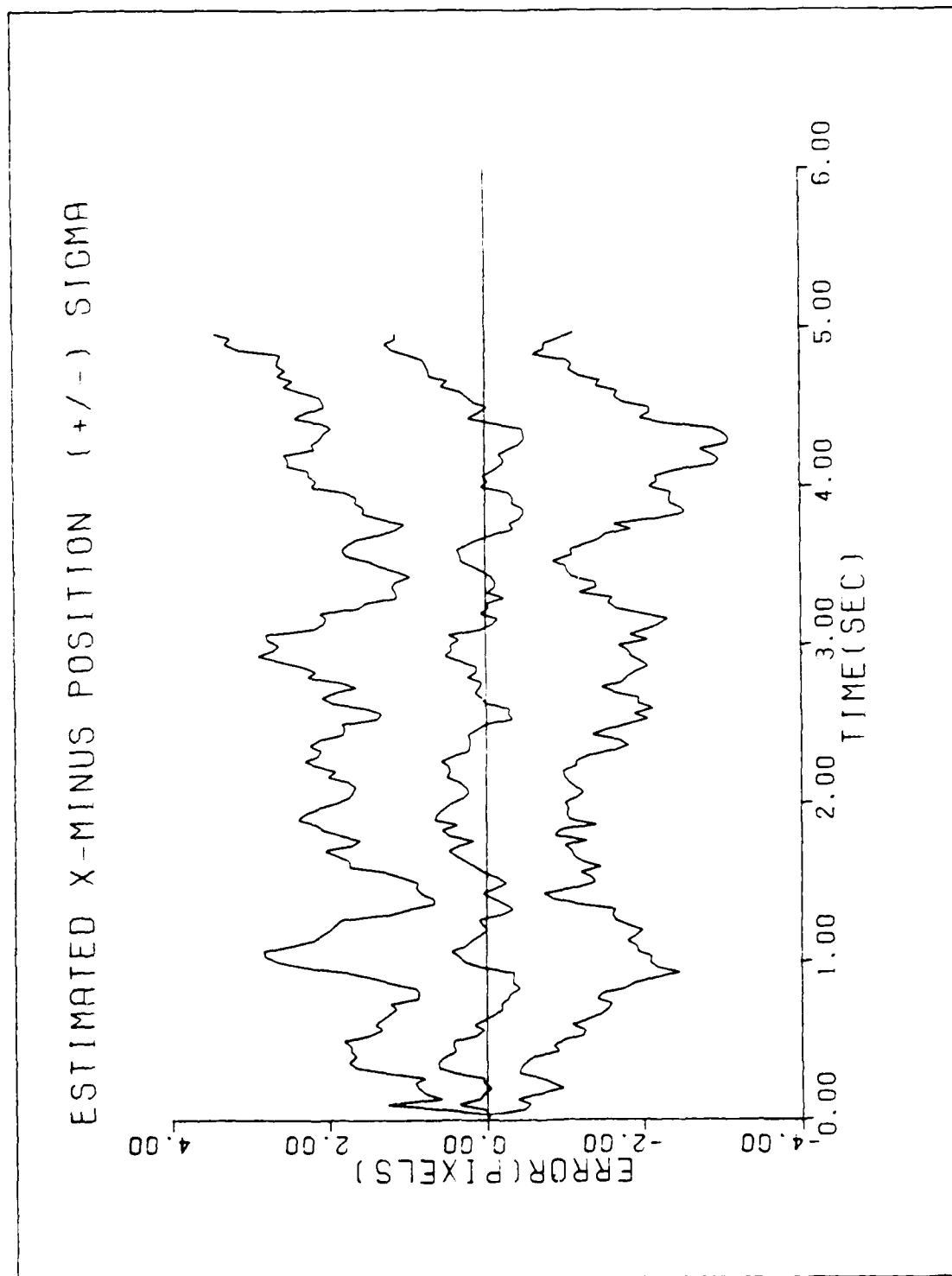


Figure E-19. GM/MMAF/T2/10-G/QB2

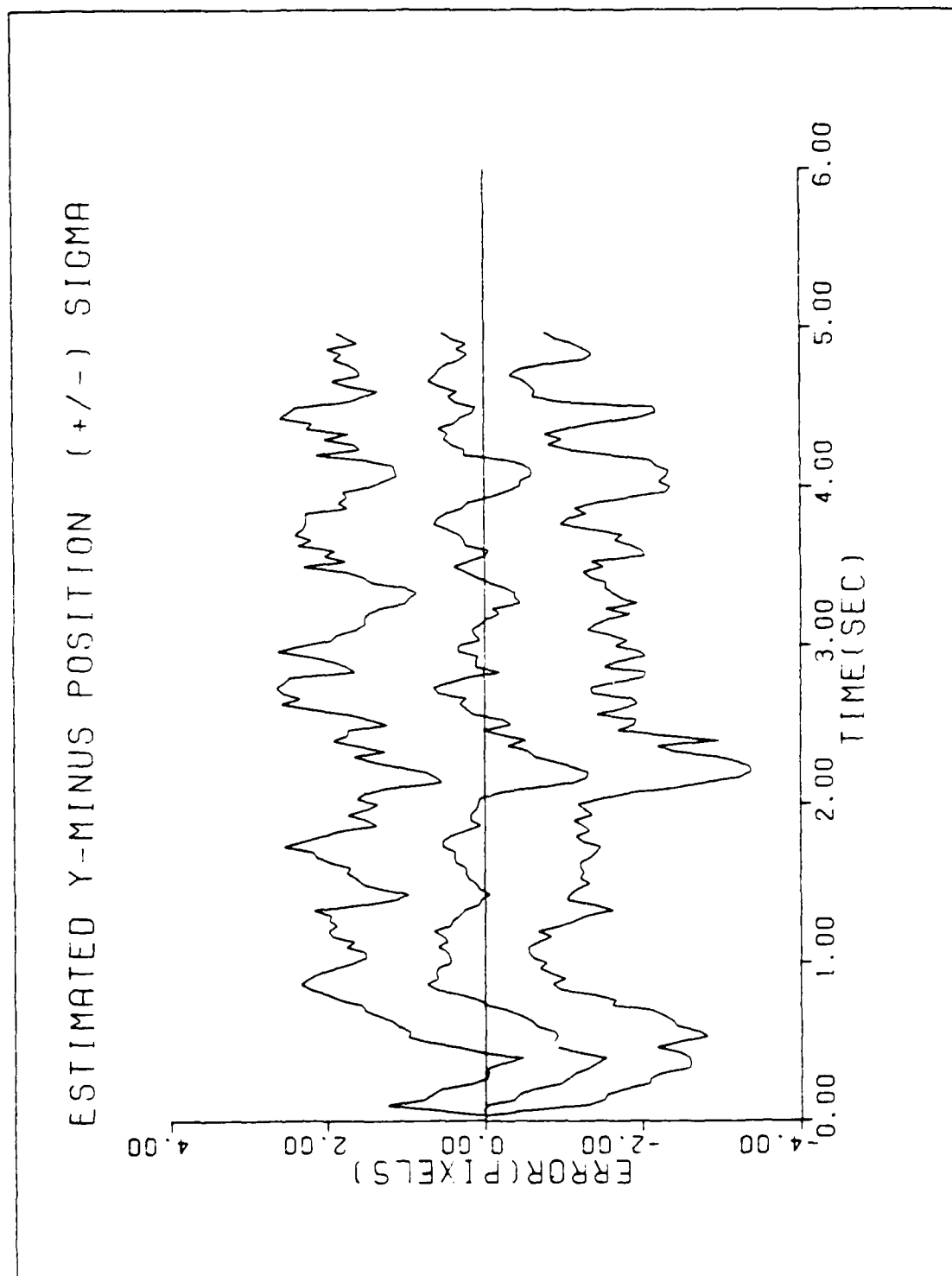


Figure E-20. GM/MMAF/T2/10-G/QB2

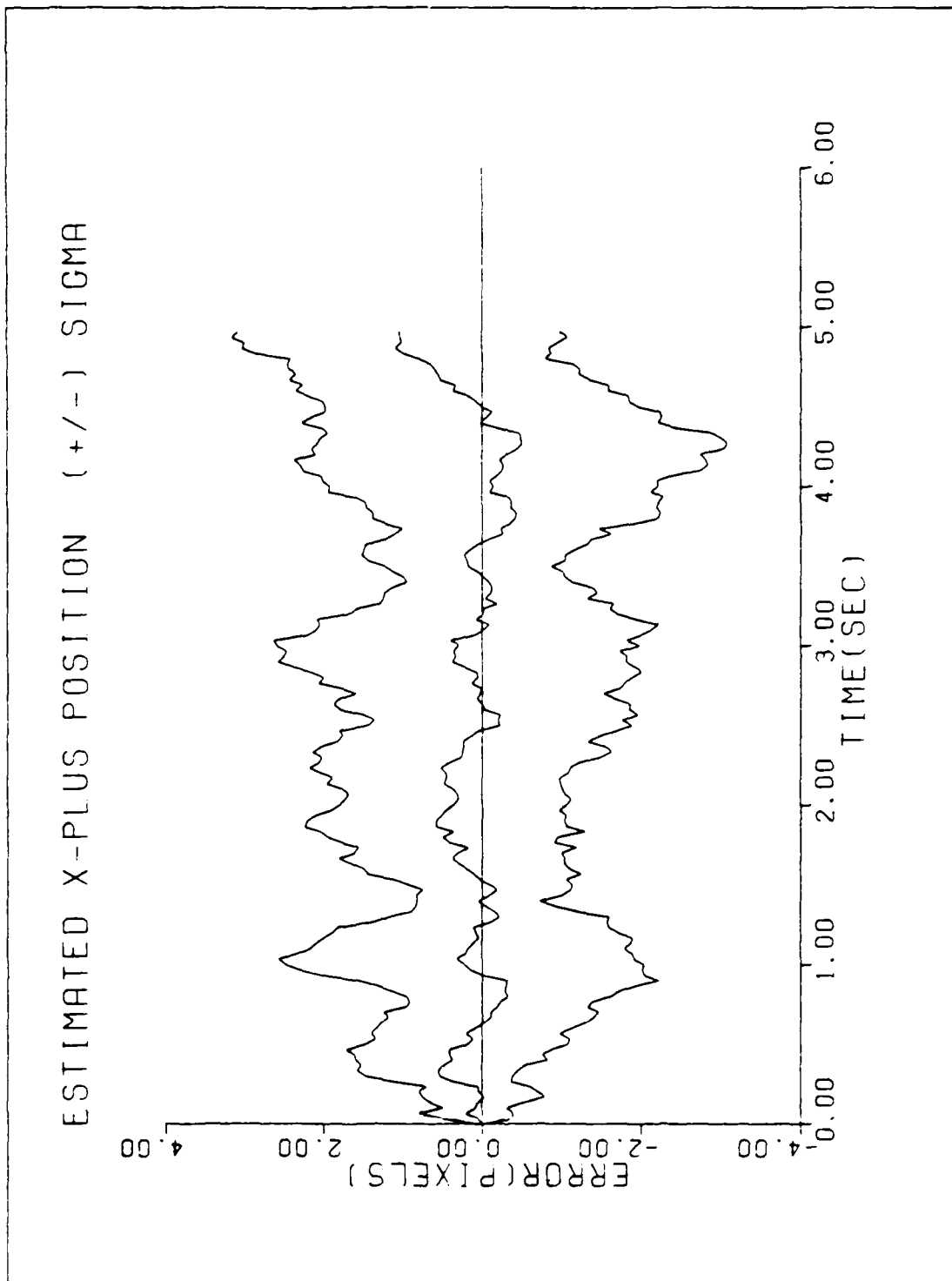


Figure E-21. GH/MMAF/T2/10-G/QB2

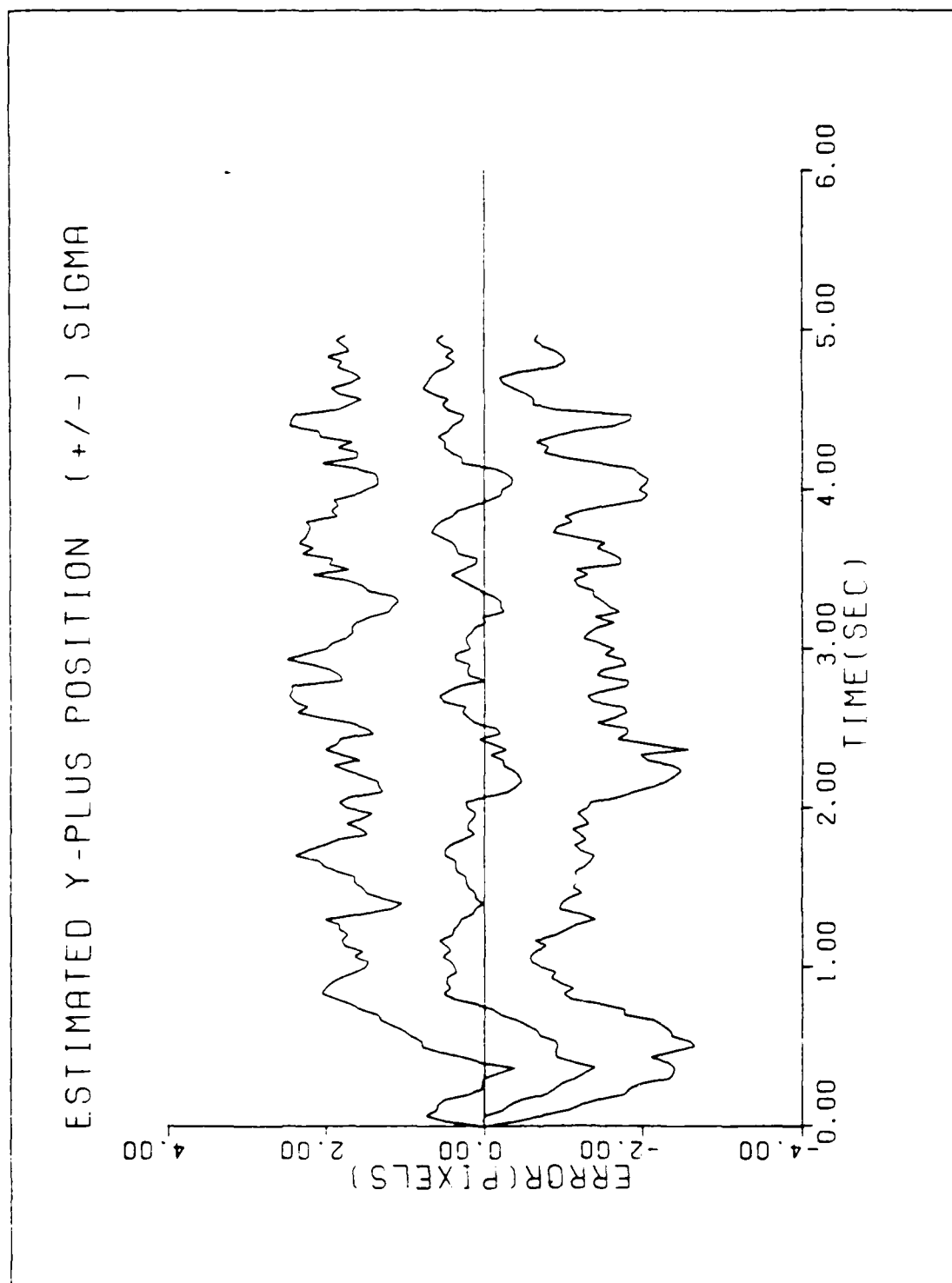


Figure E-22. GM/MNAF/T2/10-G/QB2

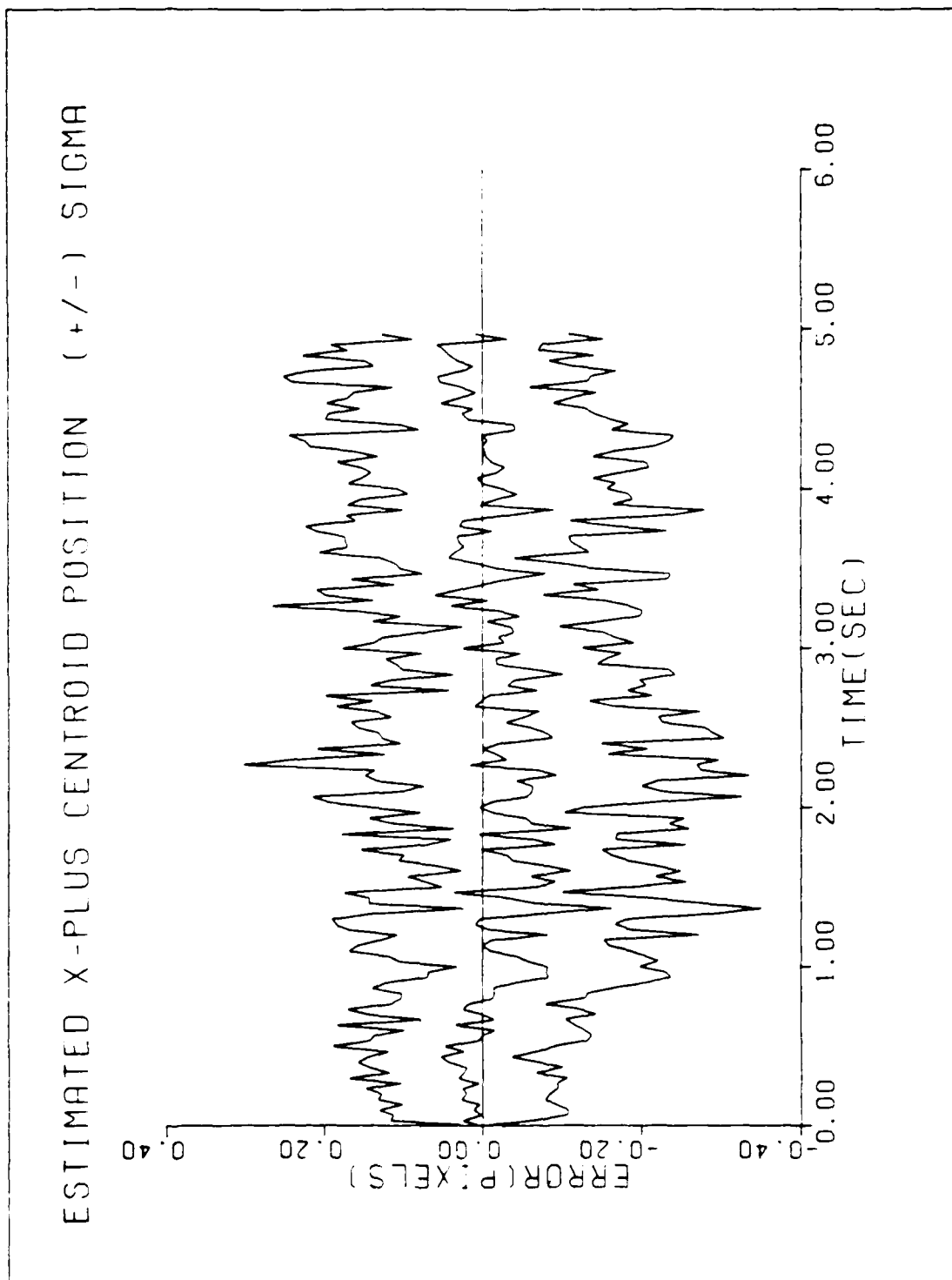


Figure E-23. GM/MMAF/T2/10-G/QB2

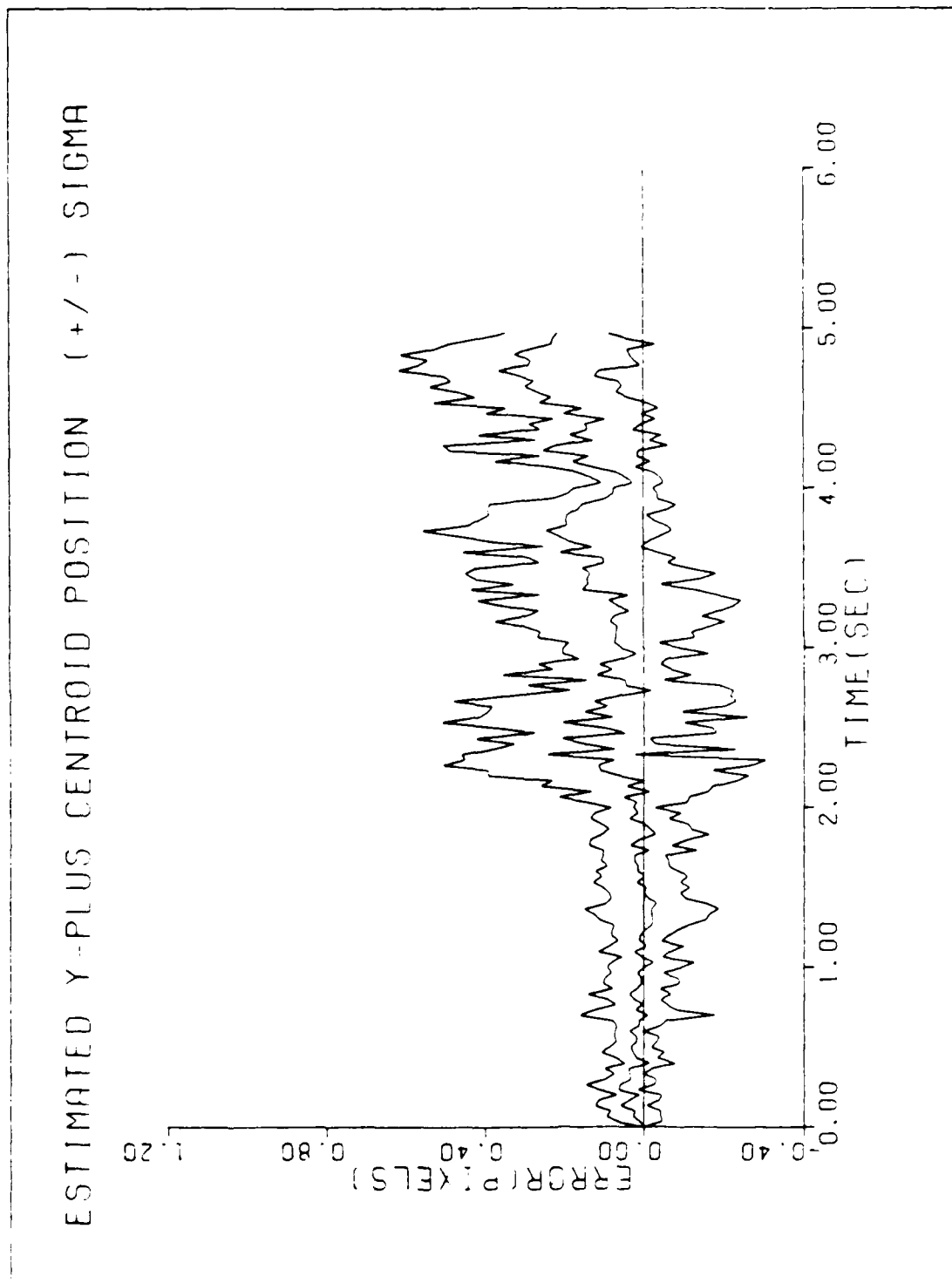


Figure E-24. GM/MMAF/T2/10-G/QB2

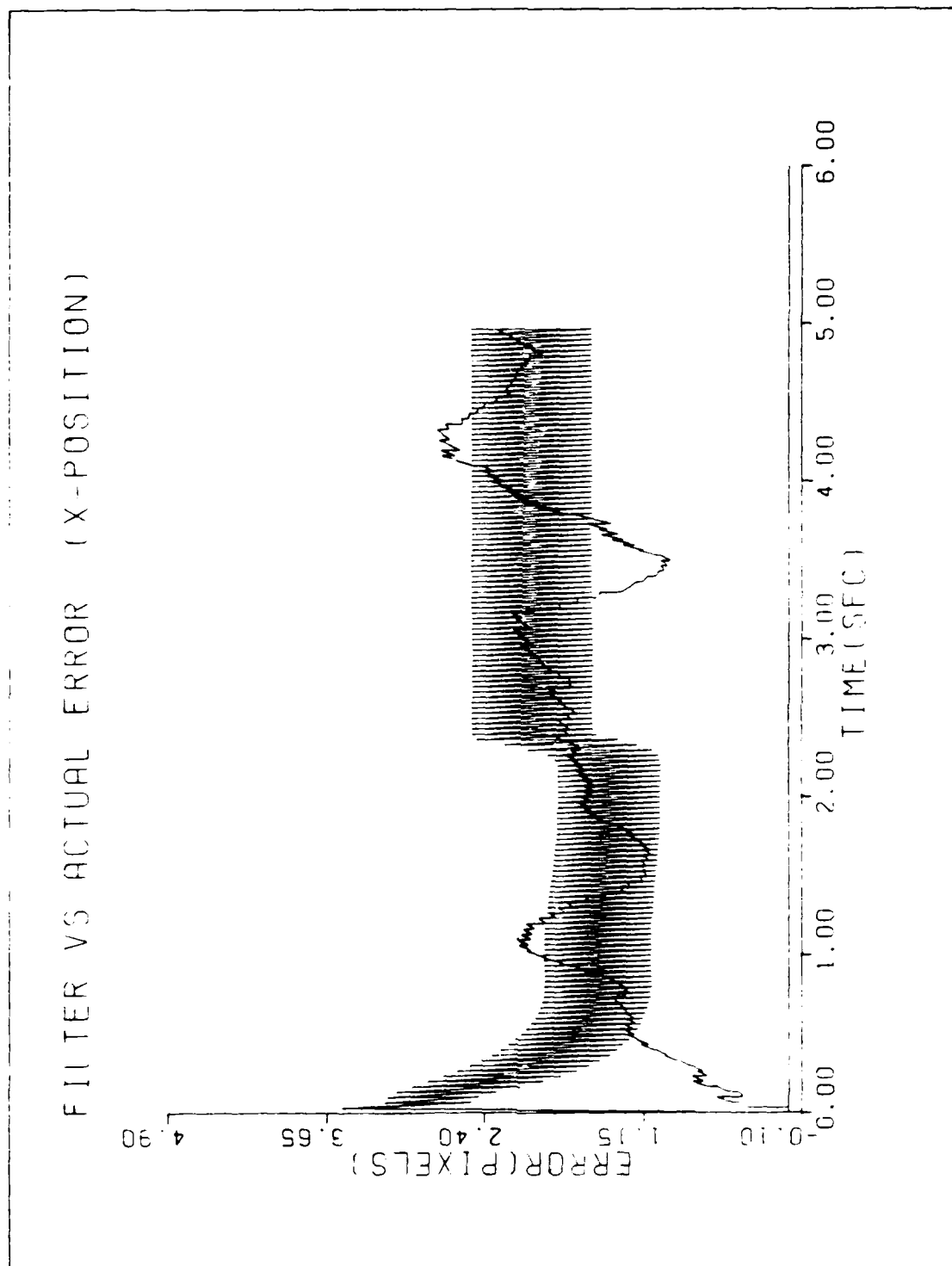


Figure E-25. GM/MMAF/T2/10-G/QB2/R+4

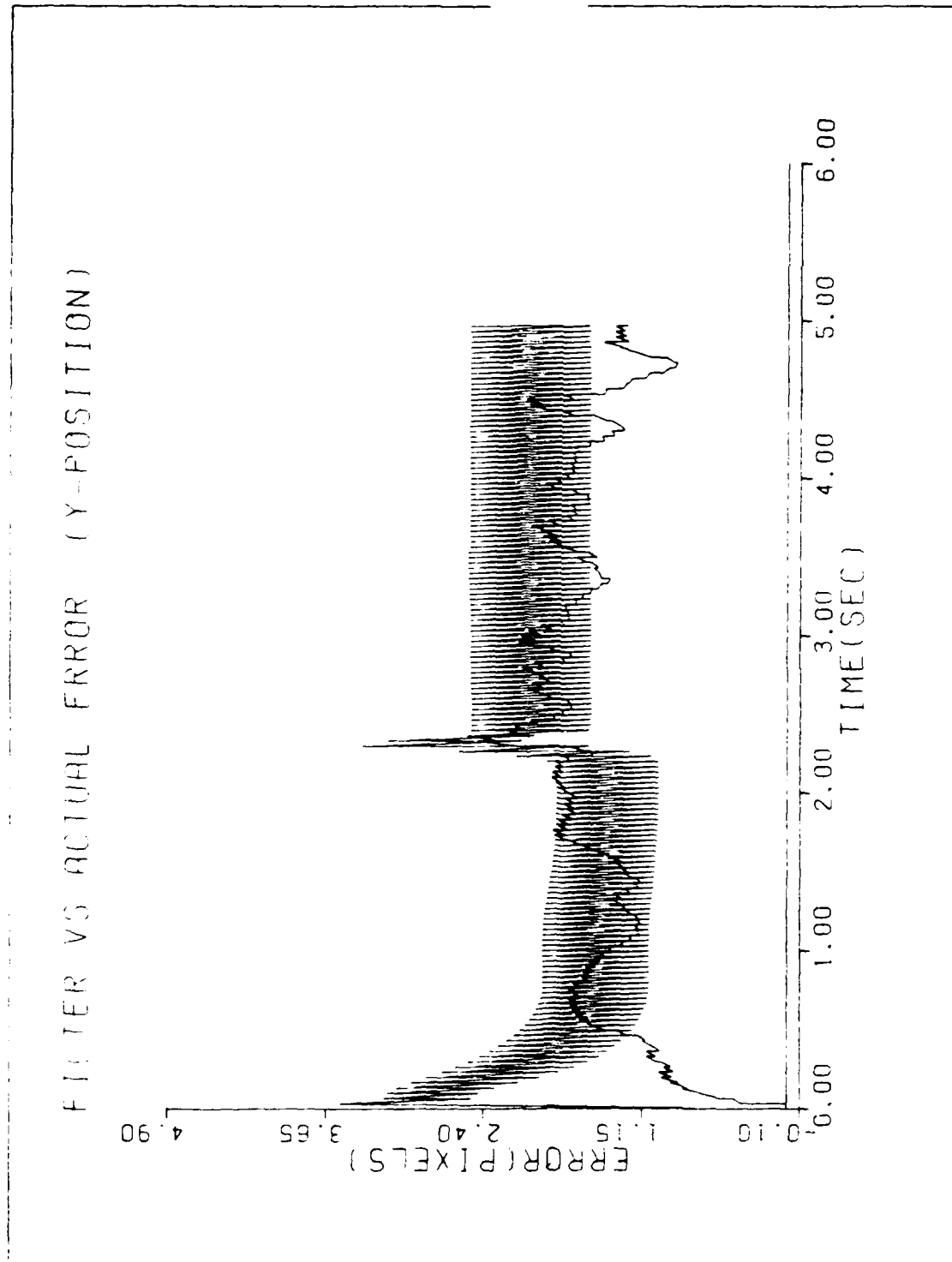


Figure E-26. GM/MMAF/T2/10-G/QB2/R+4

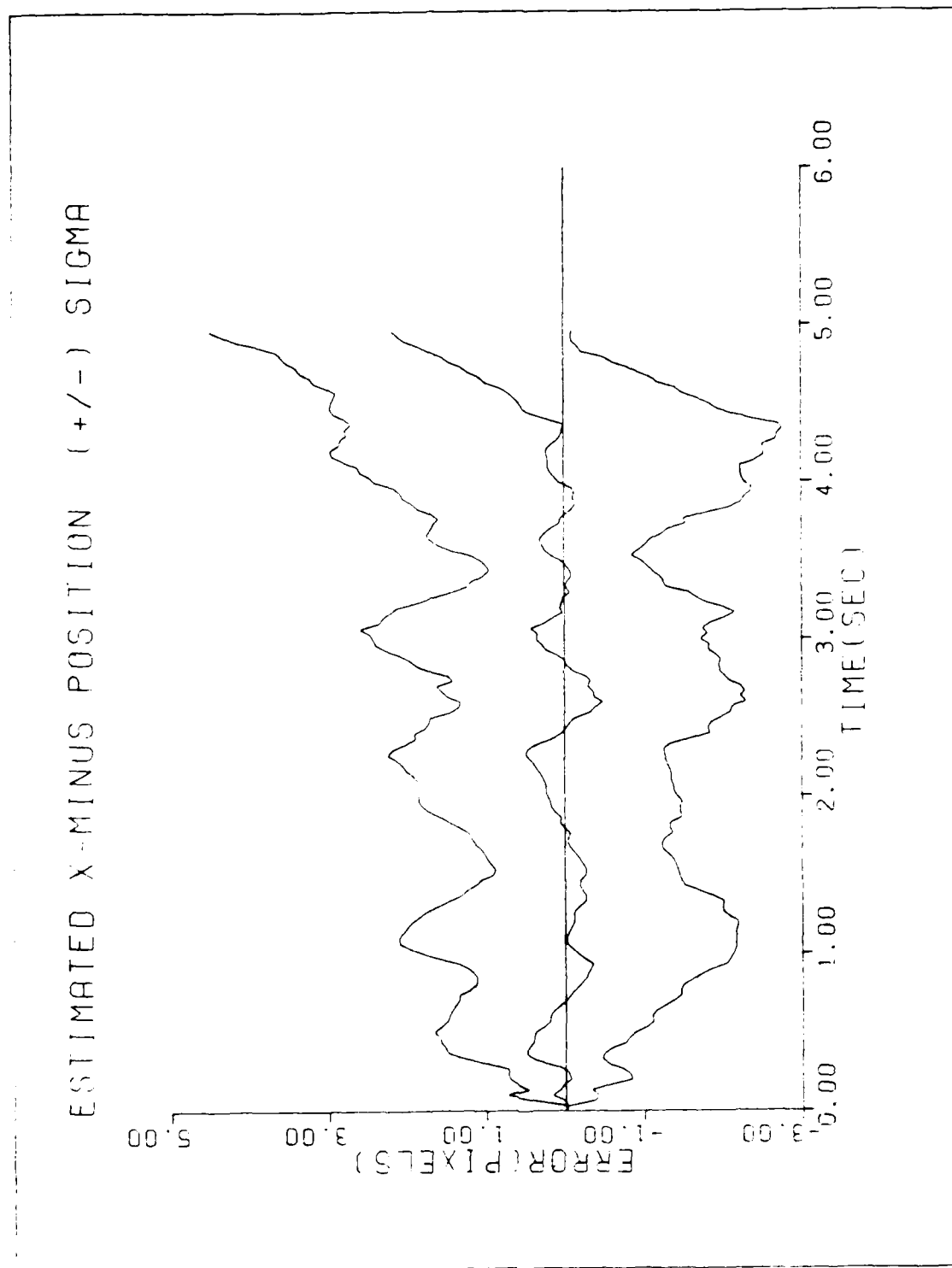


Figure E-27. GM/MMAF/T2/10-G/QB2/R+4

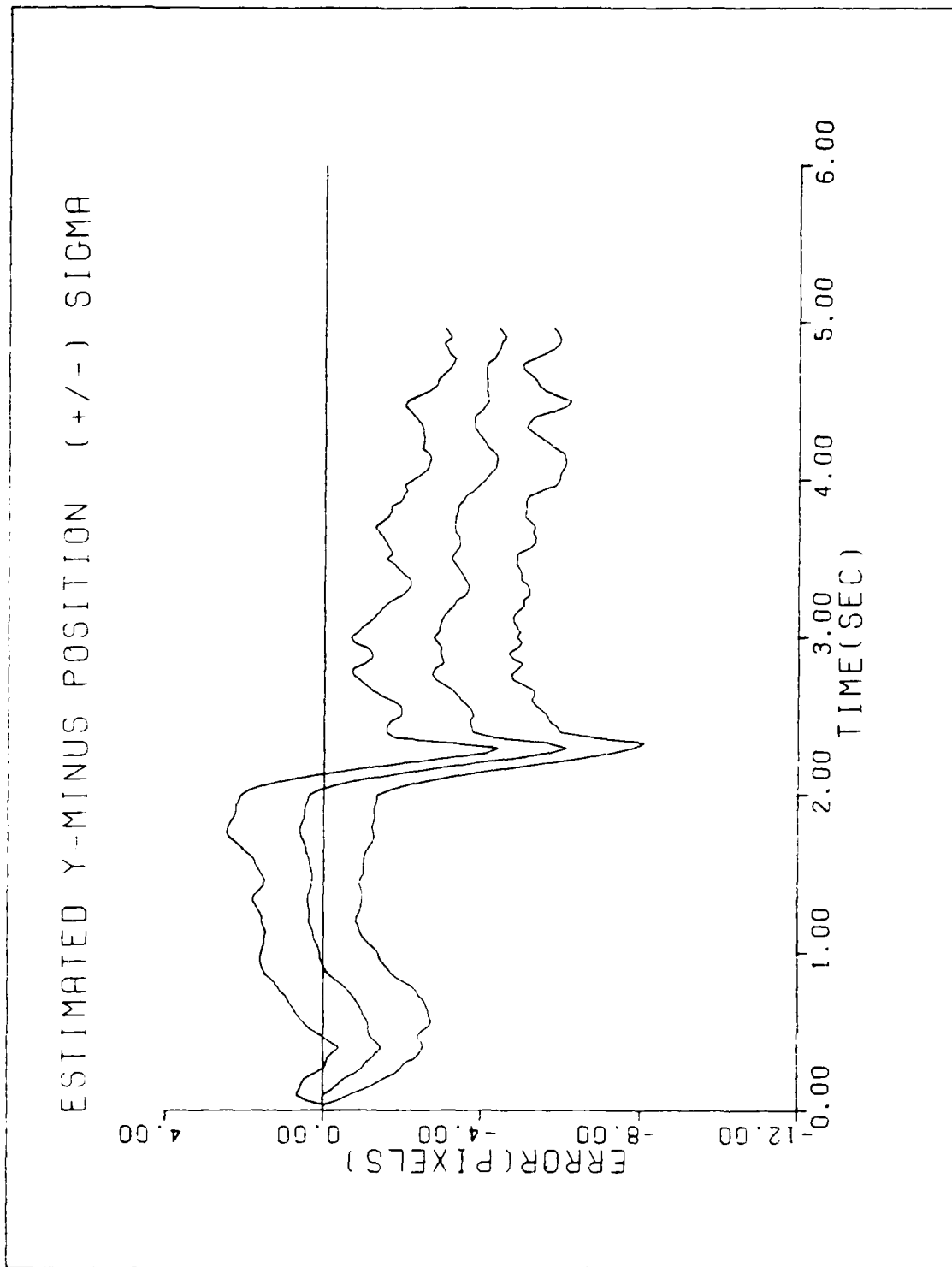


Figure E-28. GM/MMAF/T2/10-G/QB2/R+4

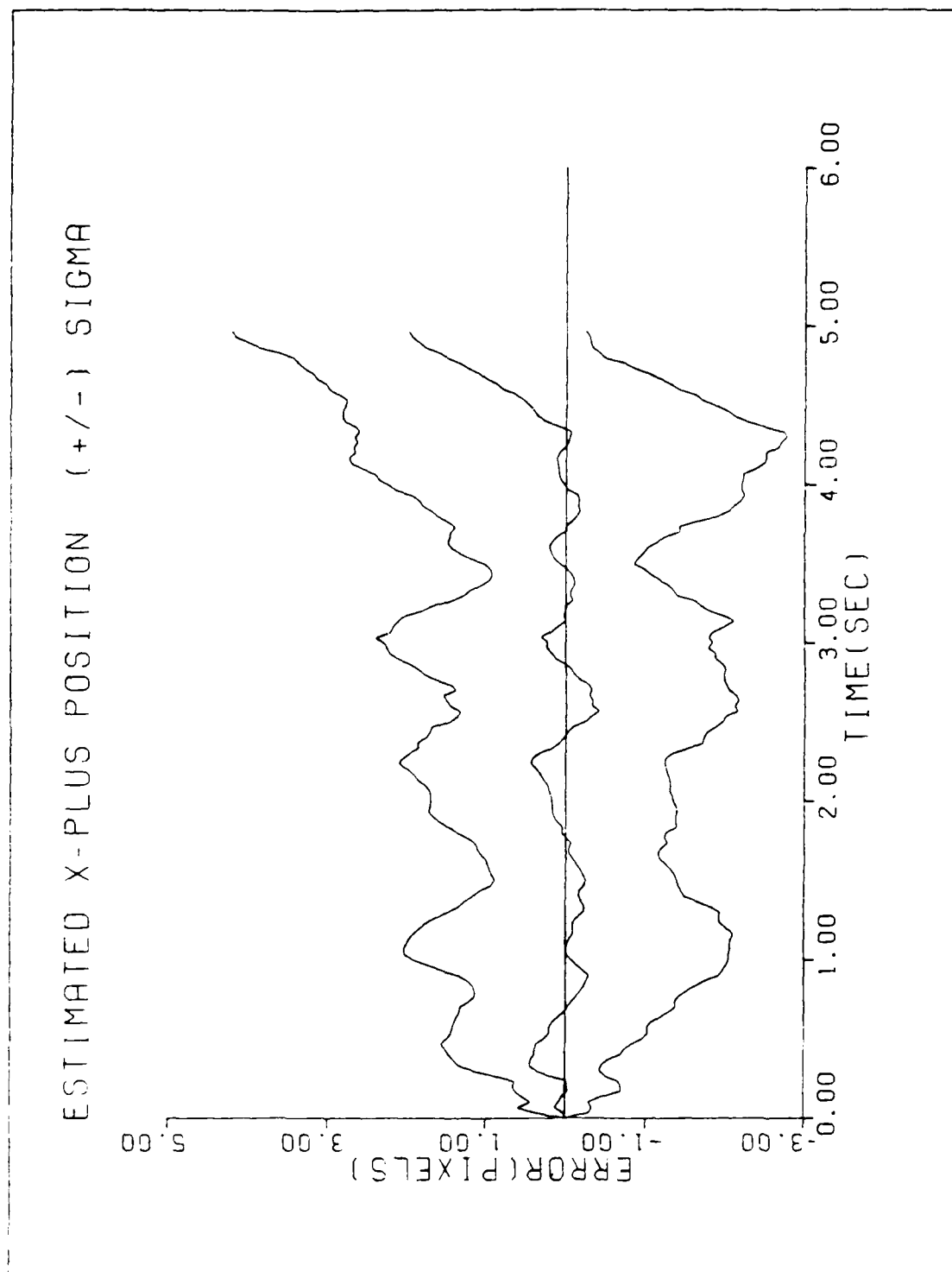


Figure E-29. GM/MMAF/T2/10-G/QB2/R+4

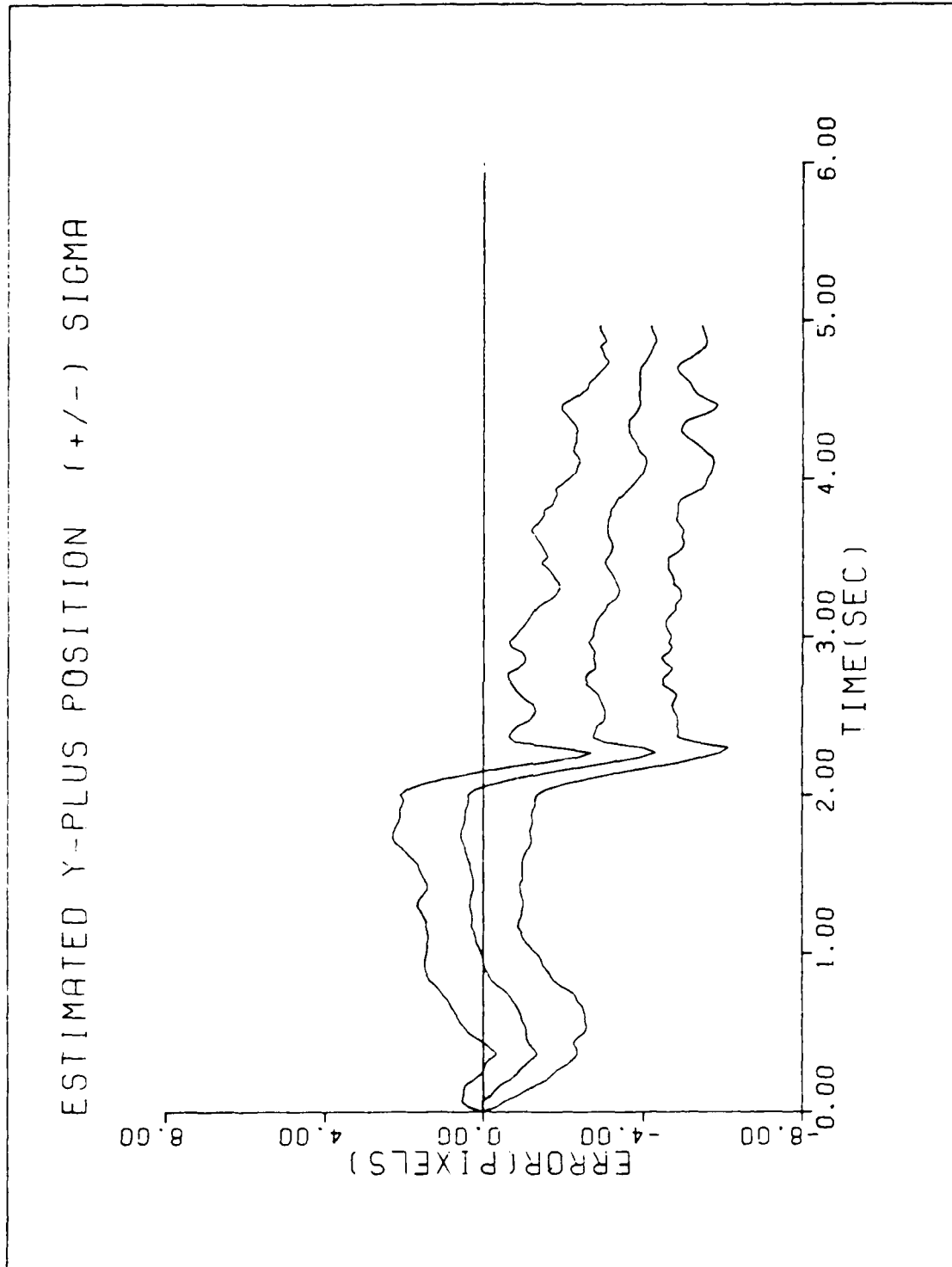


Figure E-30. GM/MMAF/T2/10-G/QB2/R+4

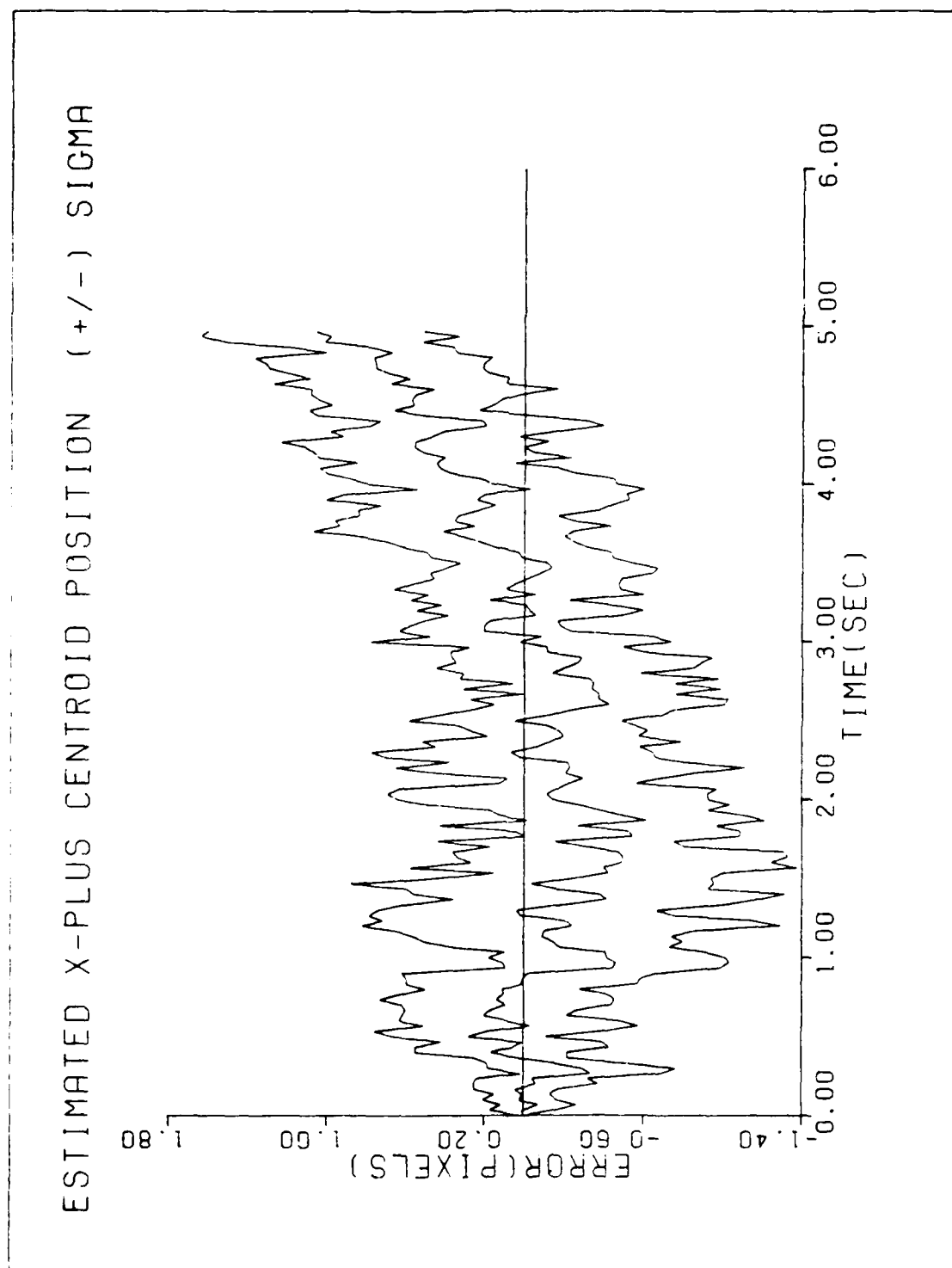


Figure E-31. GM/MAF/T2/10-G/QB2/R+4

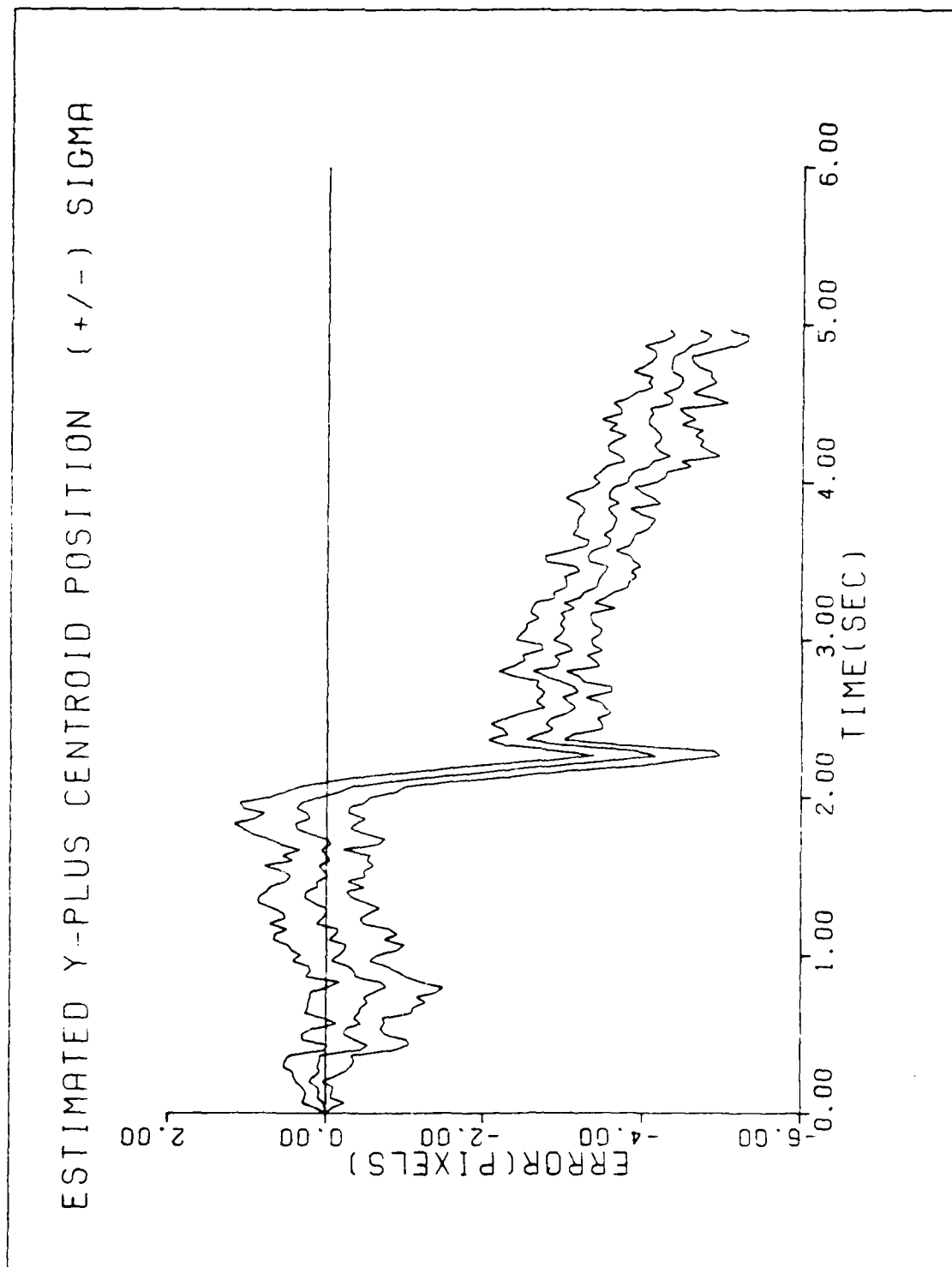


Figure E-32. GM/MMAF/T2/10-G/QB2/R+4

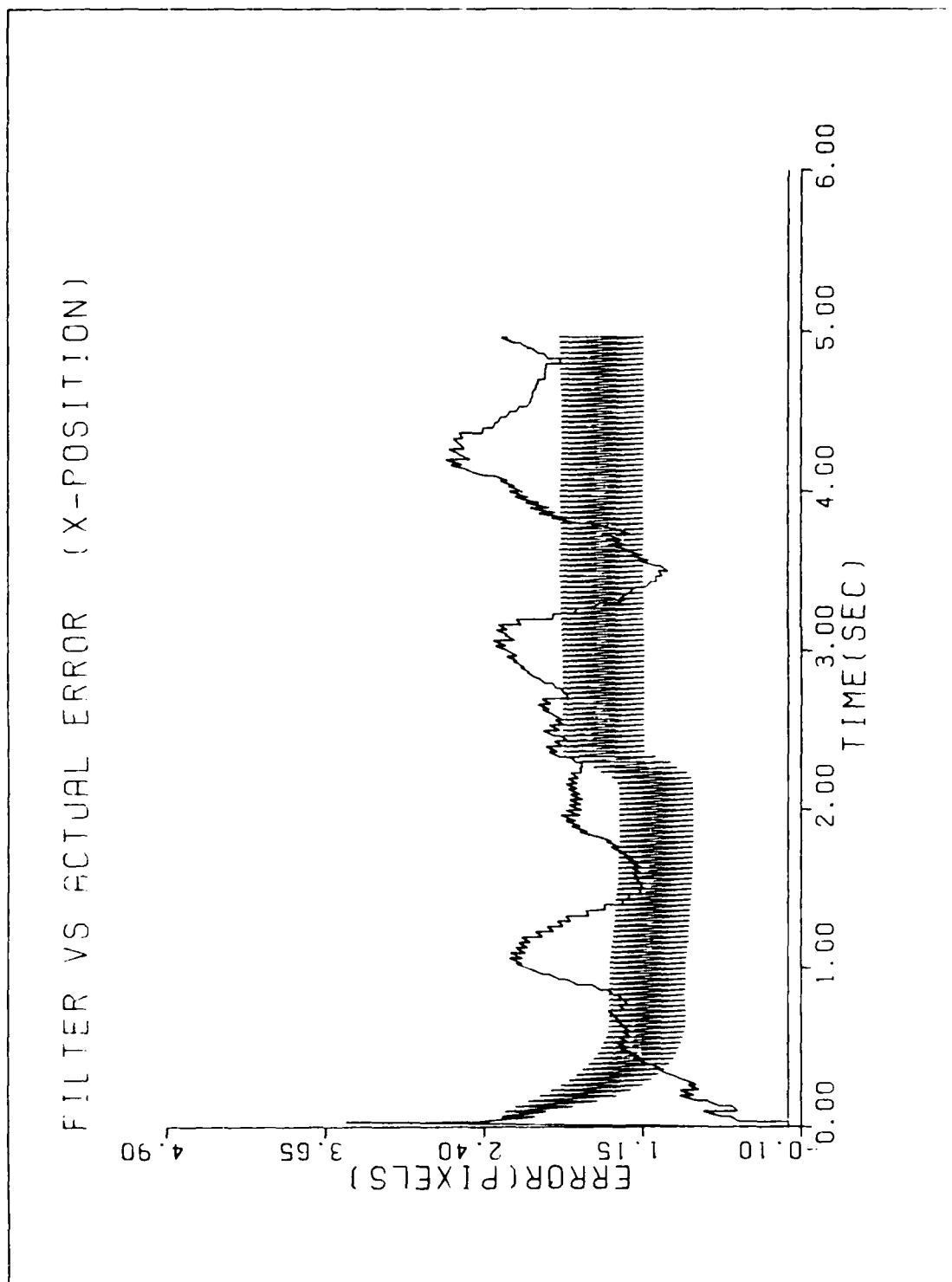


Figure E-33. GM/MAF/T2/10-G/QB2/R+1.5

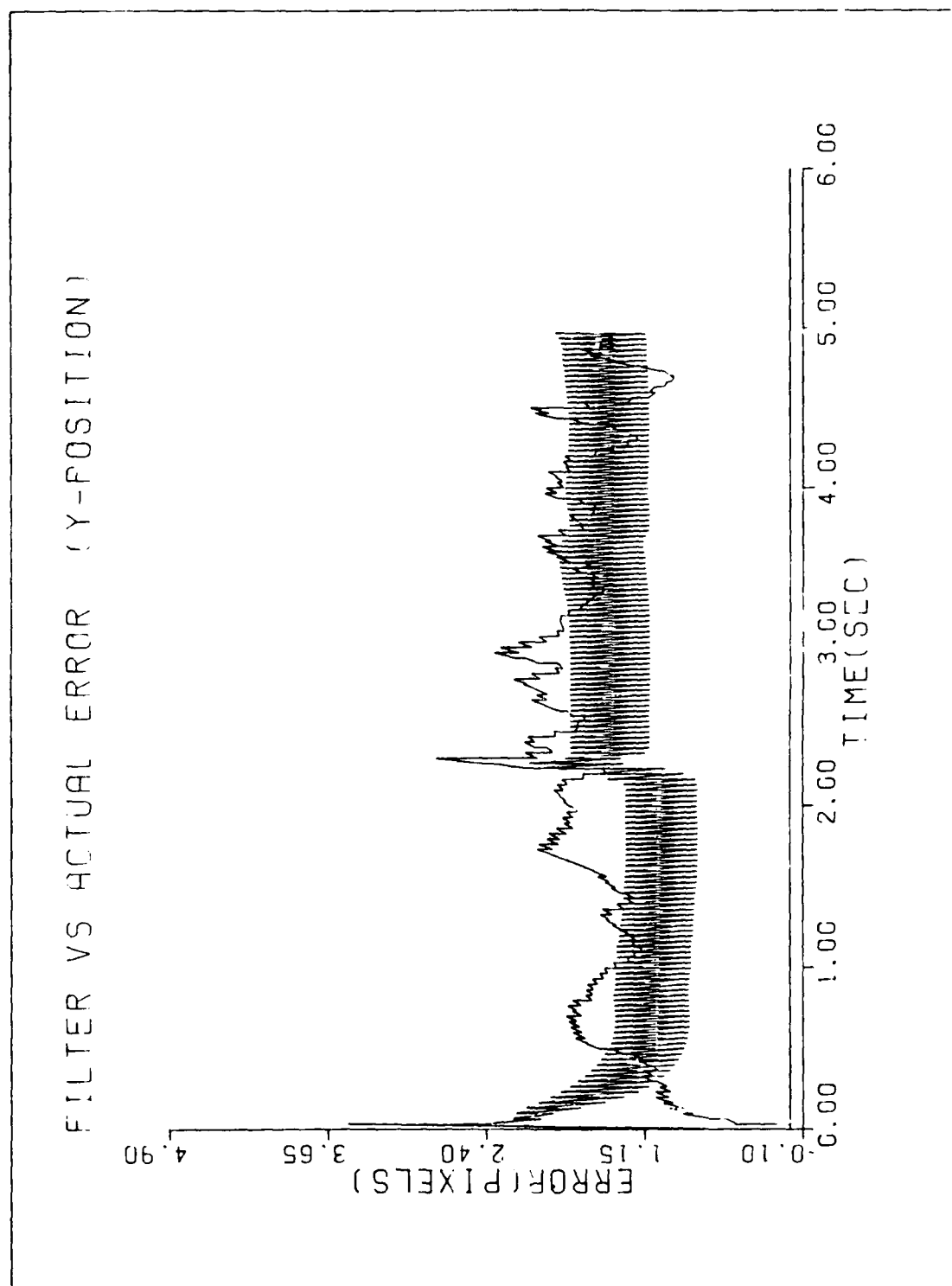


Figure E-34. GH/MMAF/T2/10-G/QB2/R+1.5

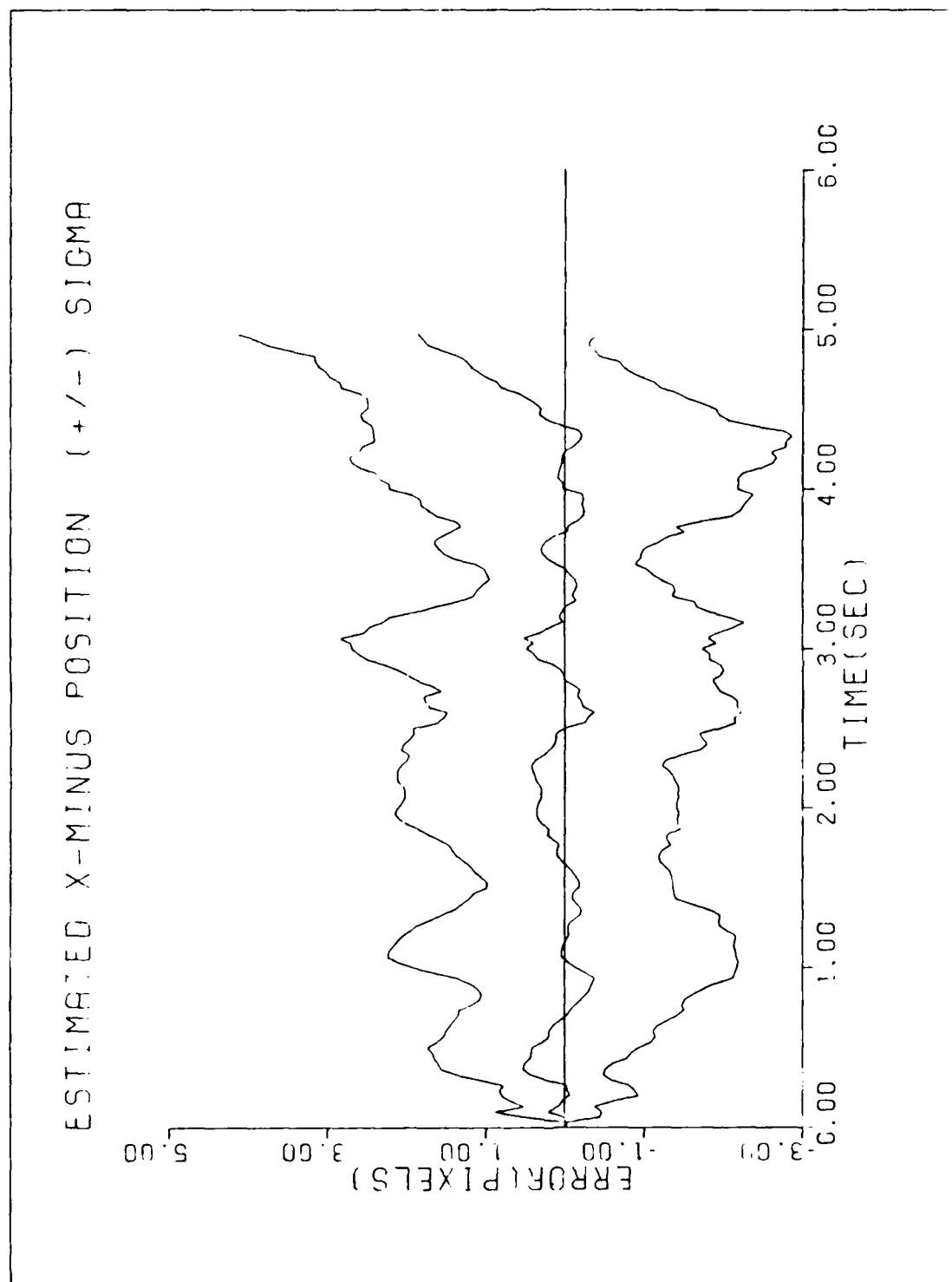


Figure E-35. GM/MMAF/T2/10-G/QB2/R+1.5

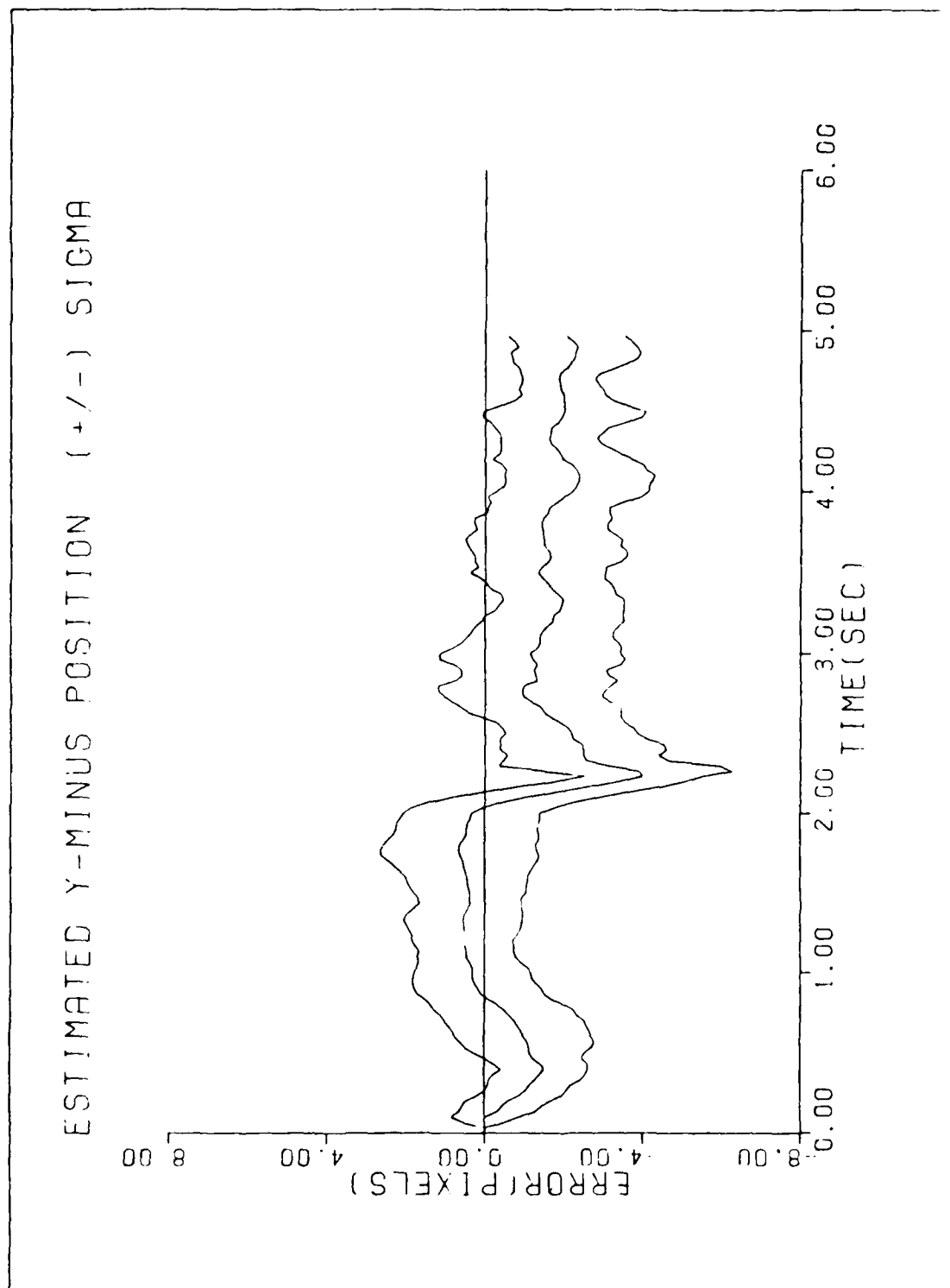


Figure E-36. GM/MAF/T2/10-G/QB2/R+1.5

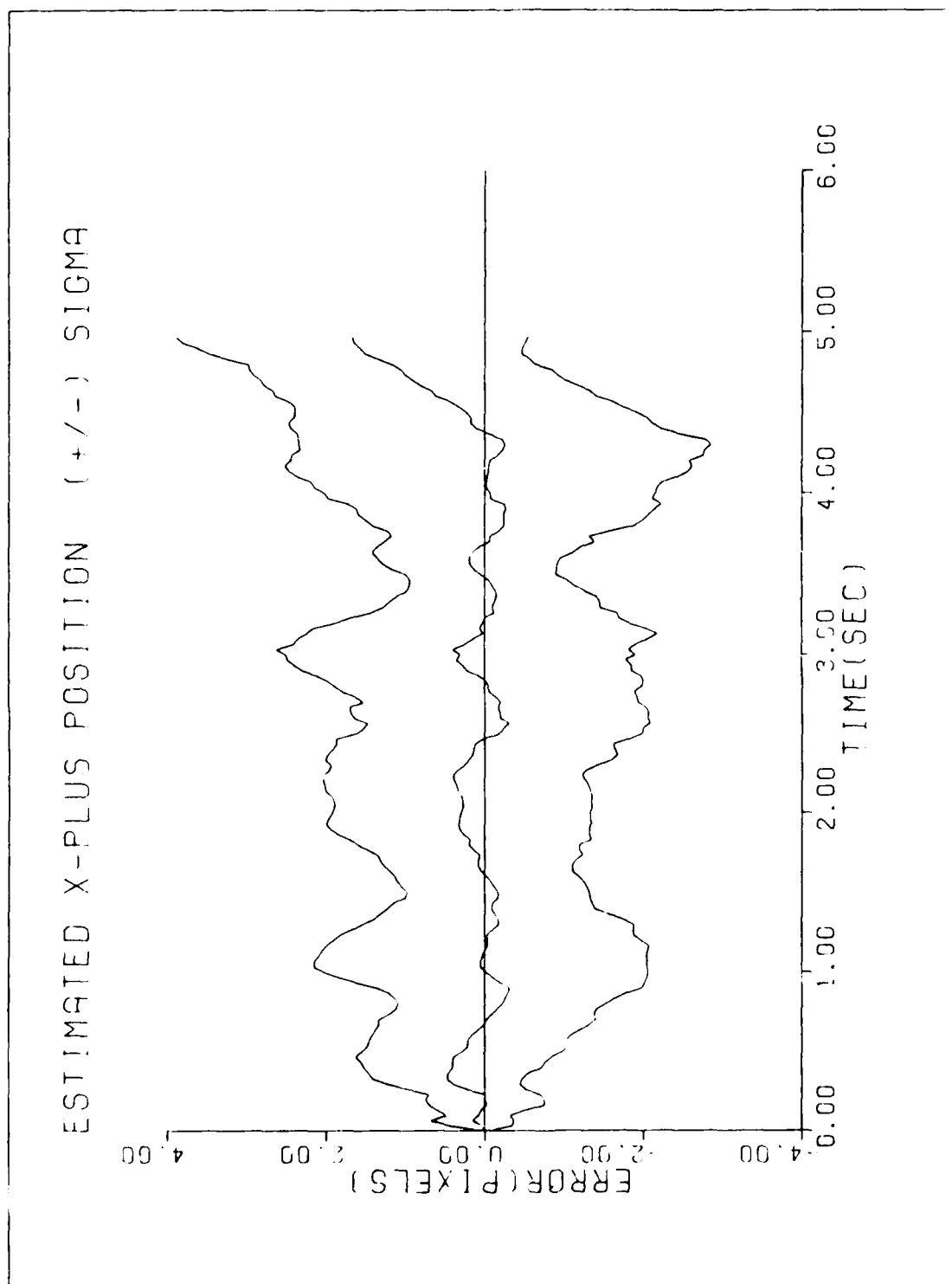


Figure E-37. G11/M1AF/T2/10-G/QB2/R+1.5

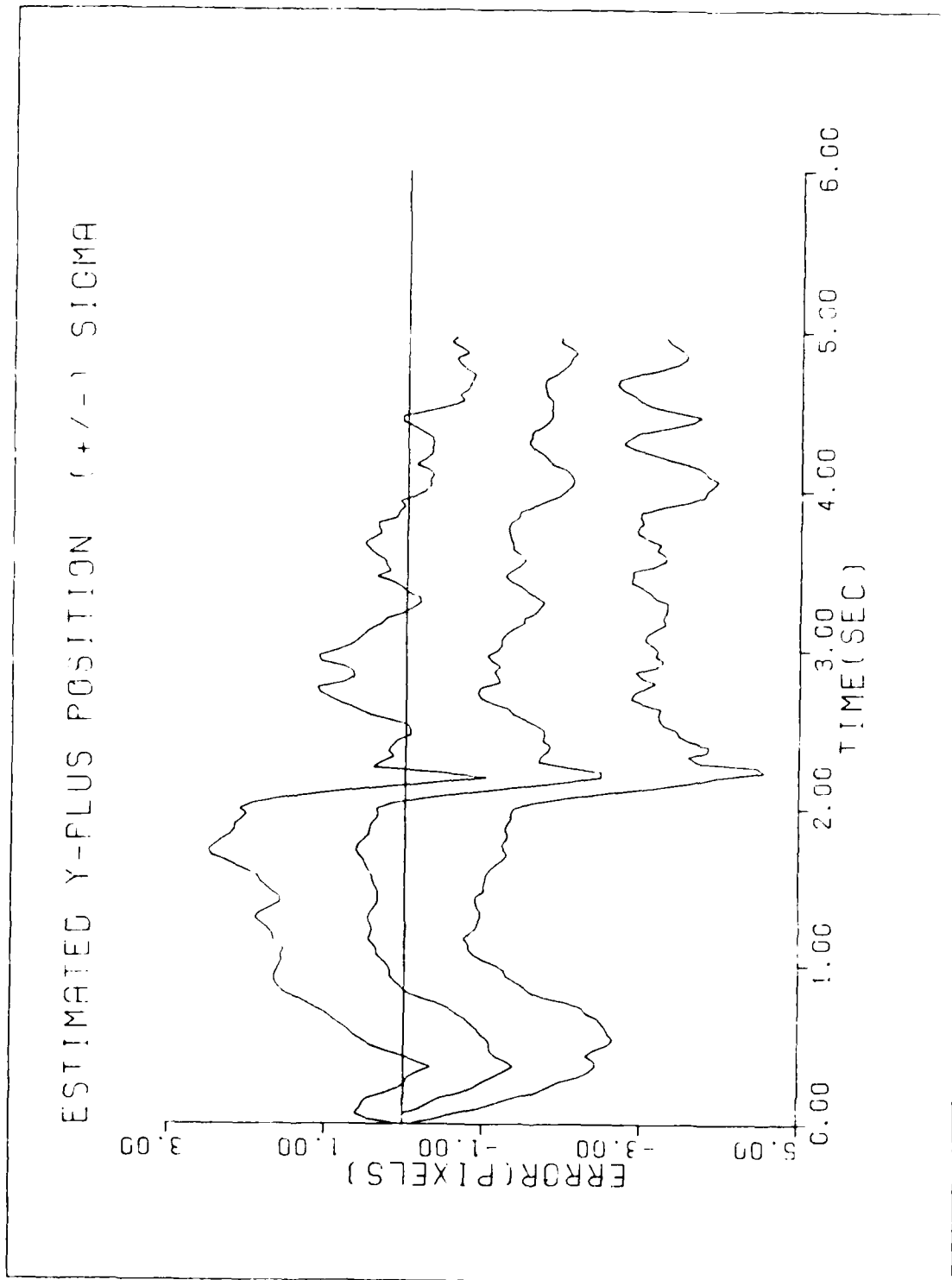


Figure E-38. GM/MMAF/T2/10-G/QB2/R+1.5

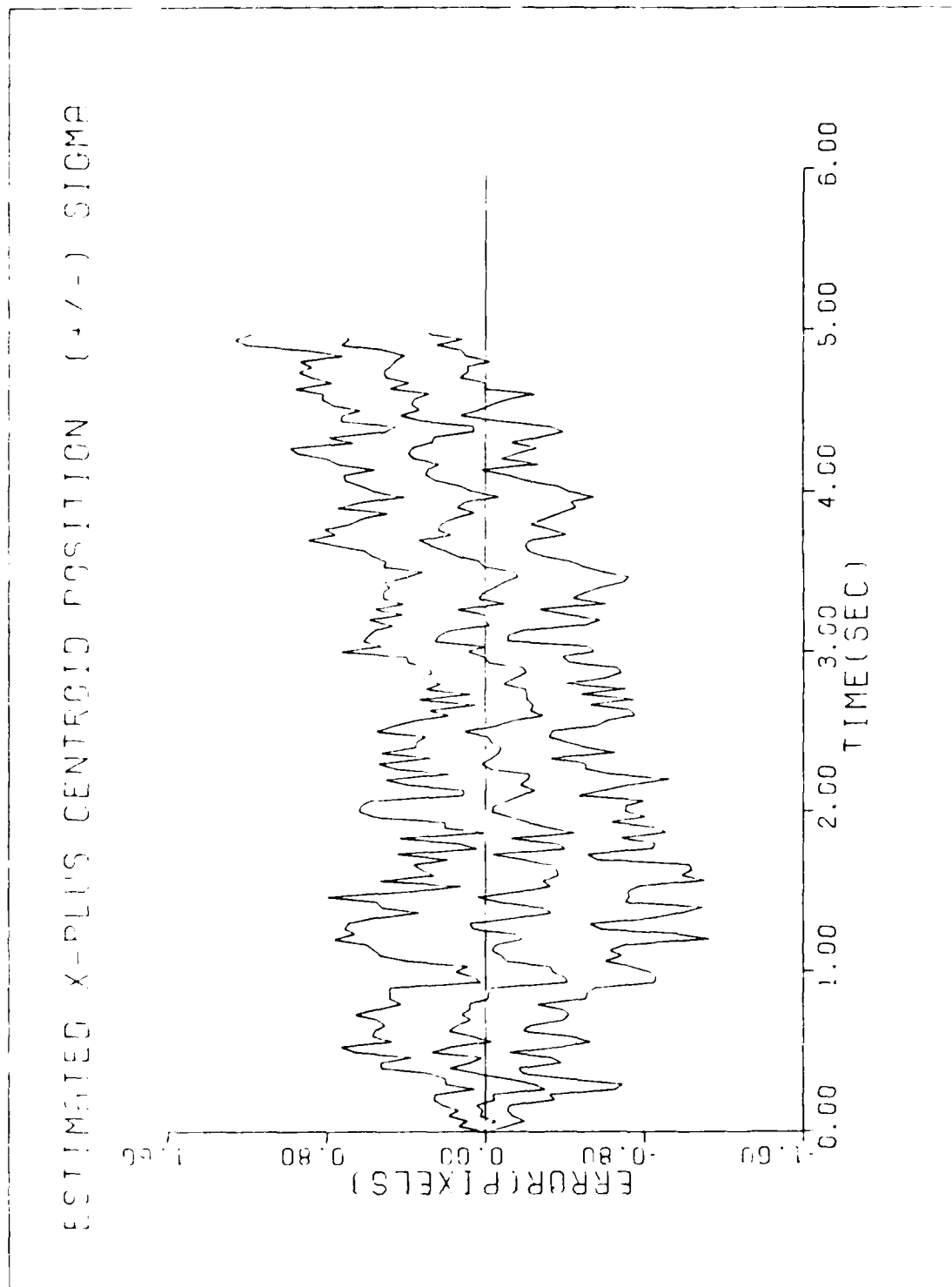


Figure E-39. GM/HMAF/T2/10-G/QB2/R+1.5

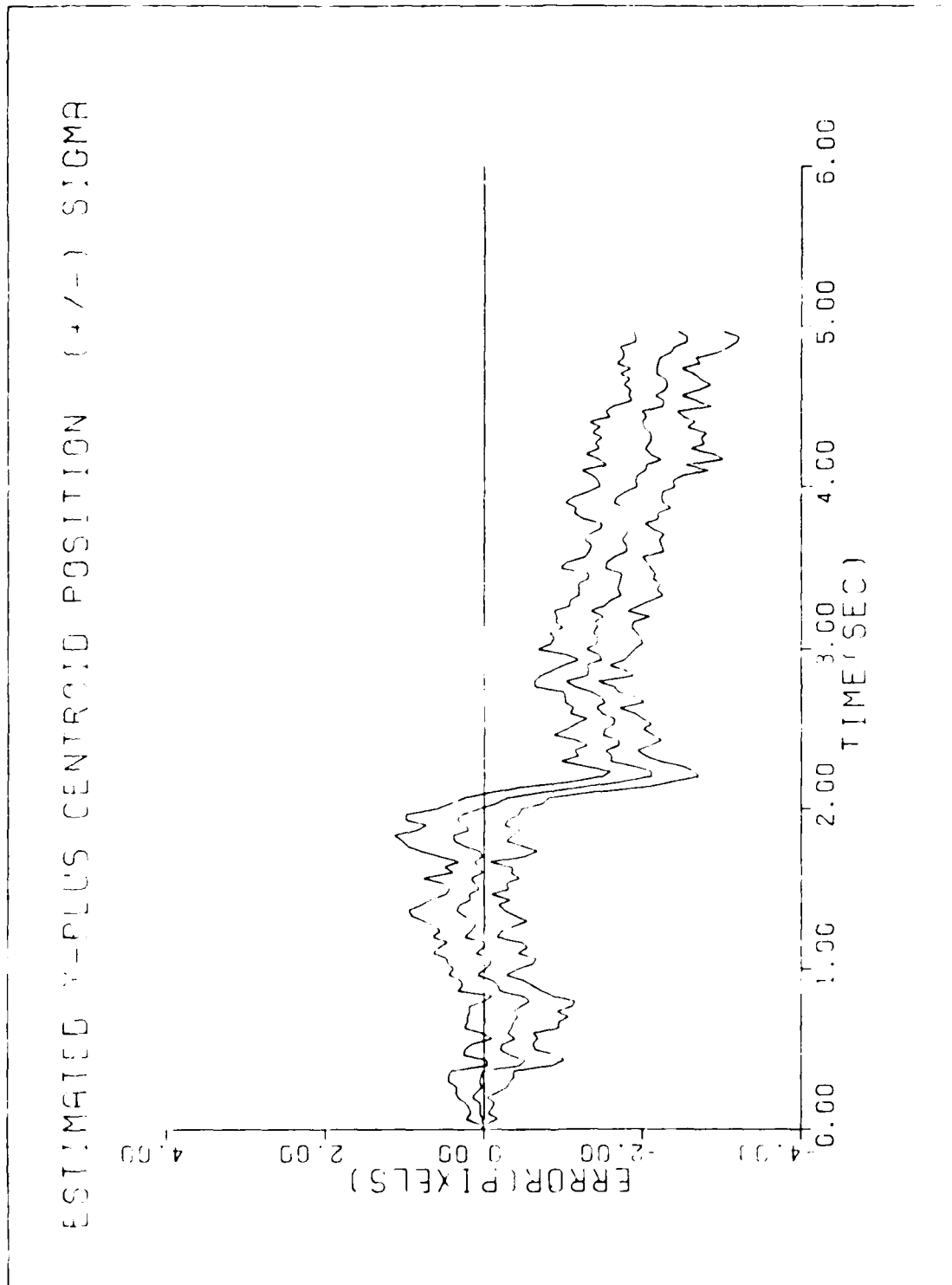


Figure E-40. GM/MAF/T2/10-G/QB2/R+1.5

APPENDIX F

MULTIPLE MODEL ADAPTIVE FILTER SIMULATIONS SCENARIO SENSITIVITY ANALYSES (REFERENCED FROM SECTION 6.5)

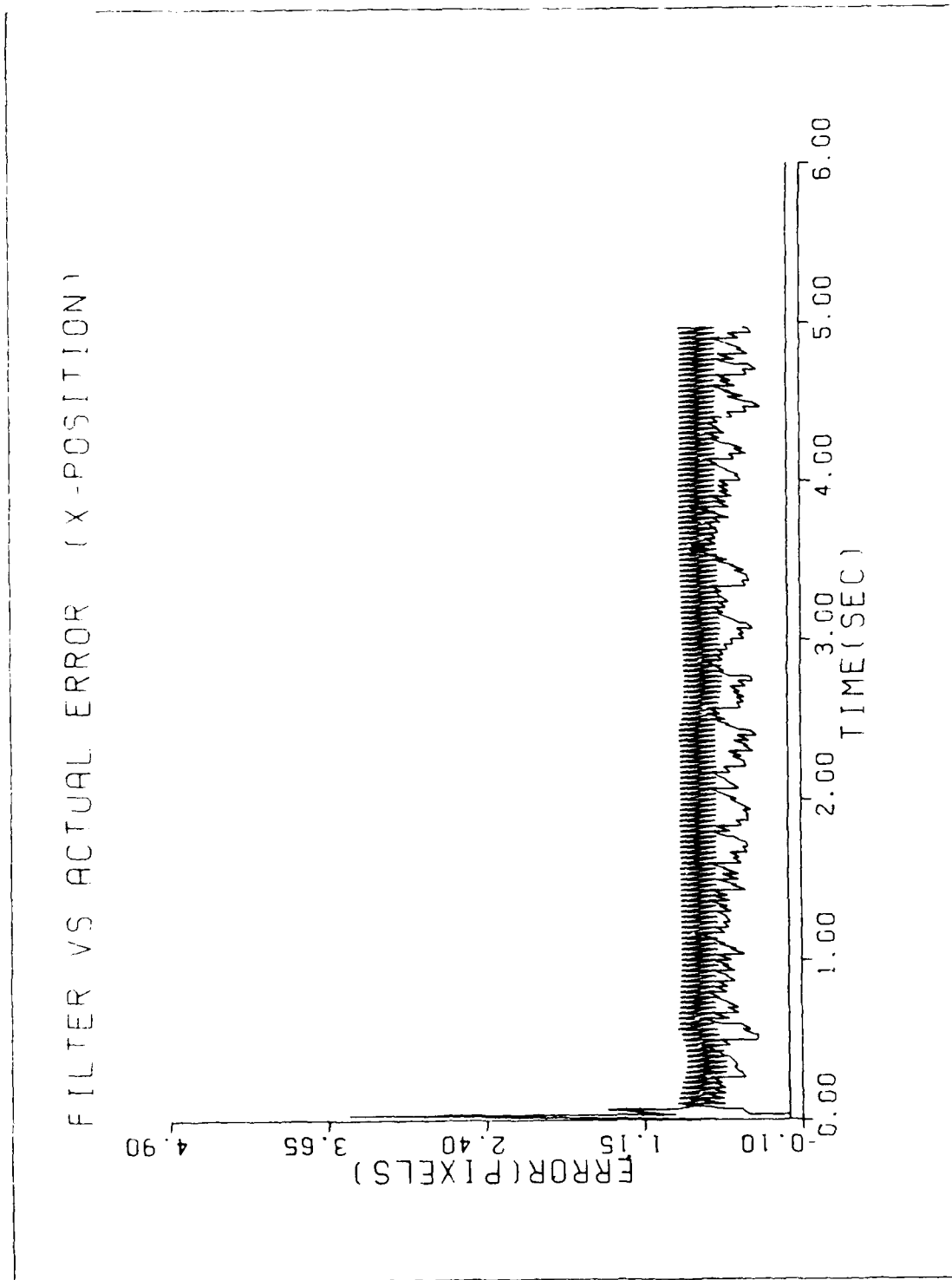


Figure F-1. $G/M/A/T/2/2-G/\tau=8,3/KP=2$ nrad

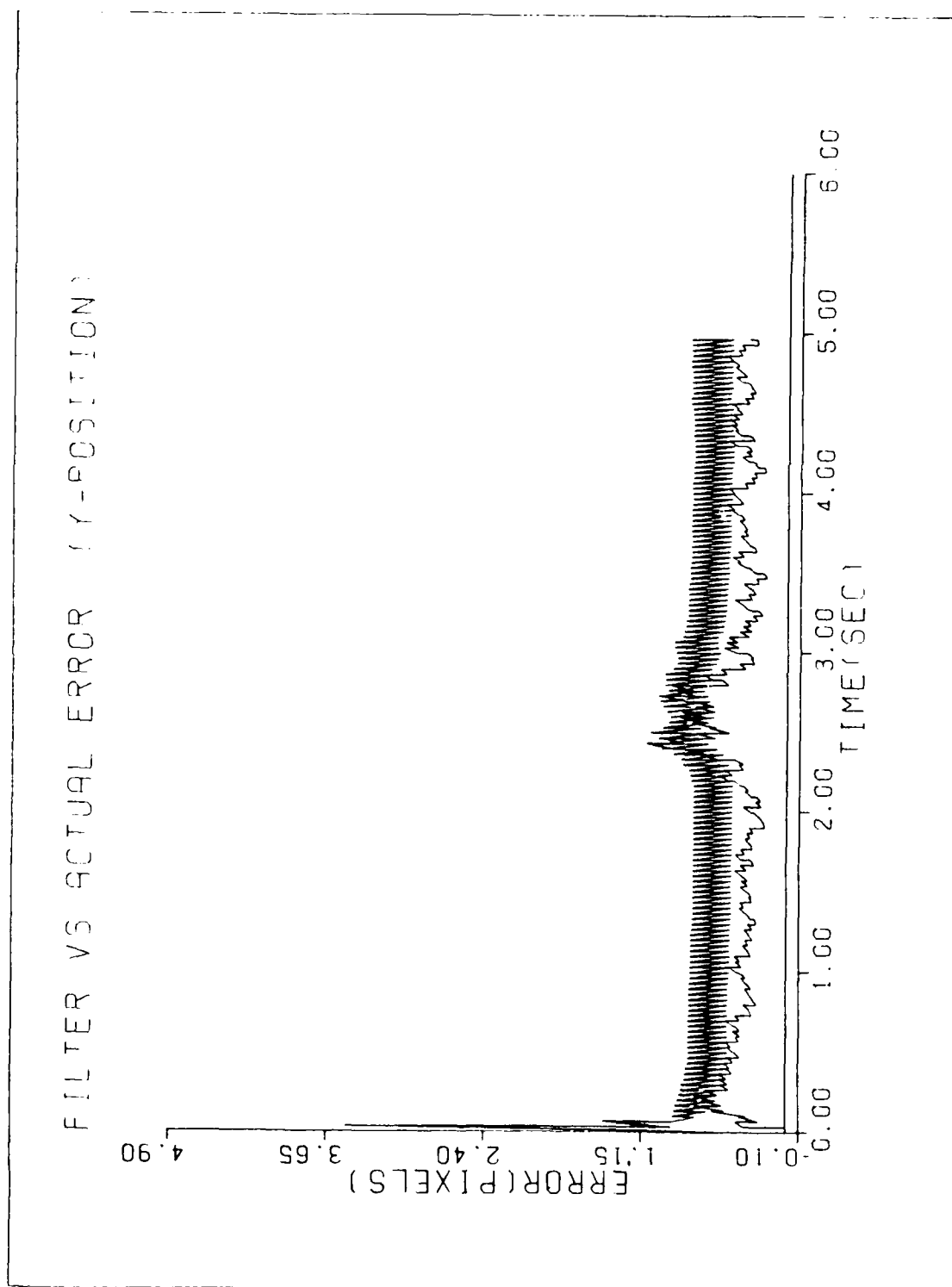


Figure F-2. $G_N/WAF/T2/2-G/\tau=8,3/KP=2$ nrad

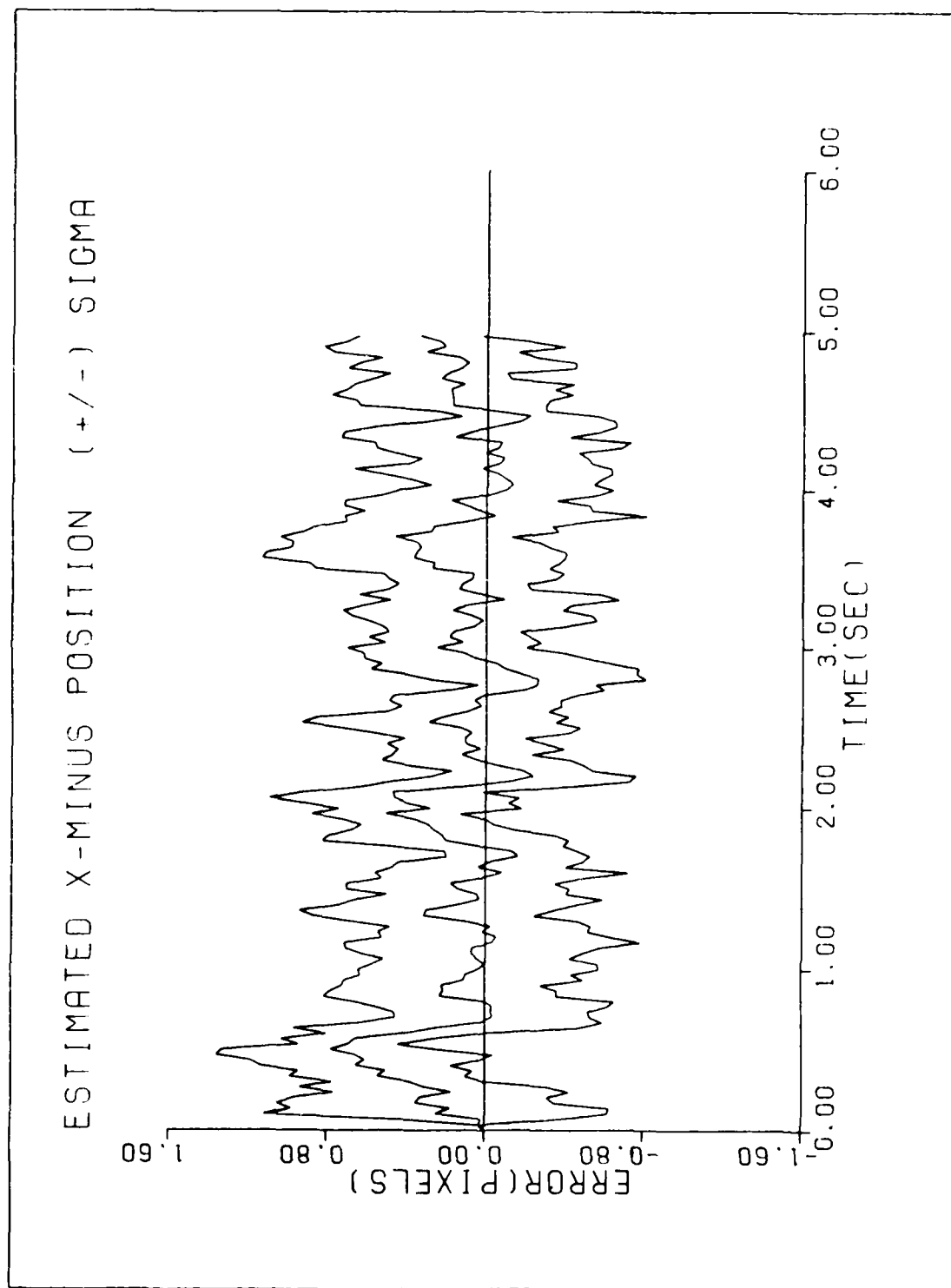


Figure F-3. $G/H/MAF/T2/2-G/T=8,3/KP=2$ nrad

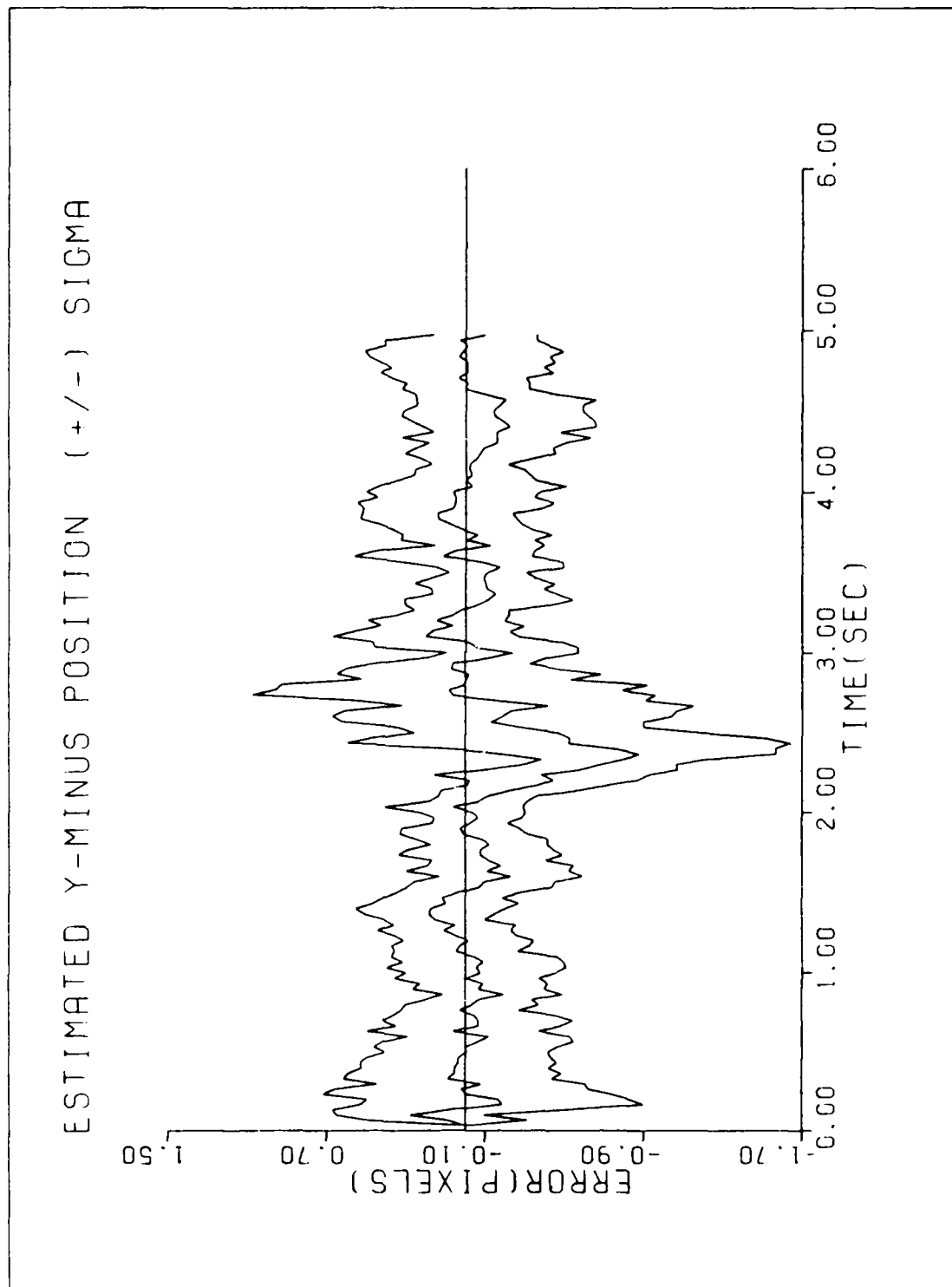


Figure F-4. $G/M/AF/T2/2-G/\tau=8,3/KP=2$ nrad

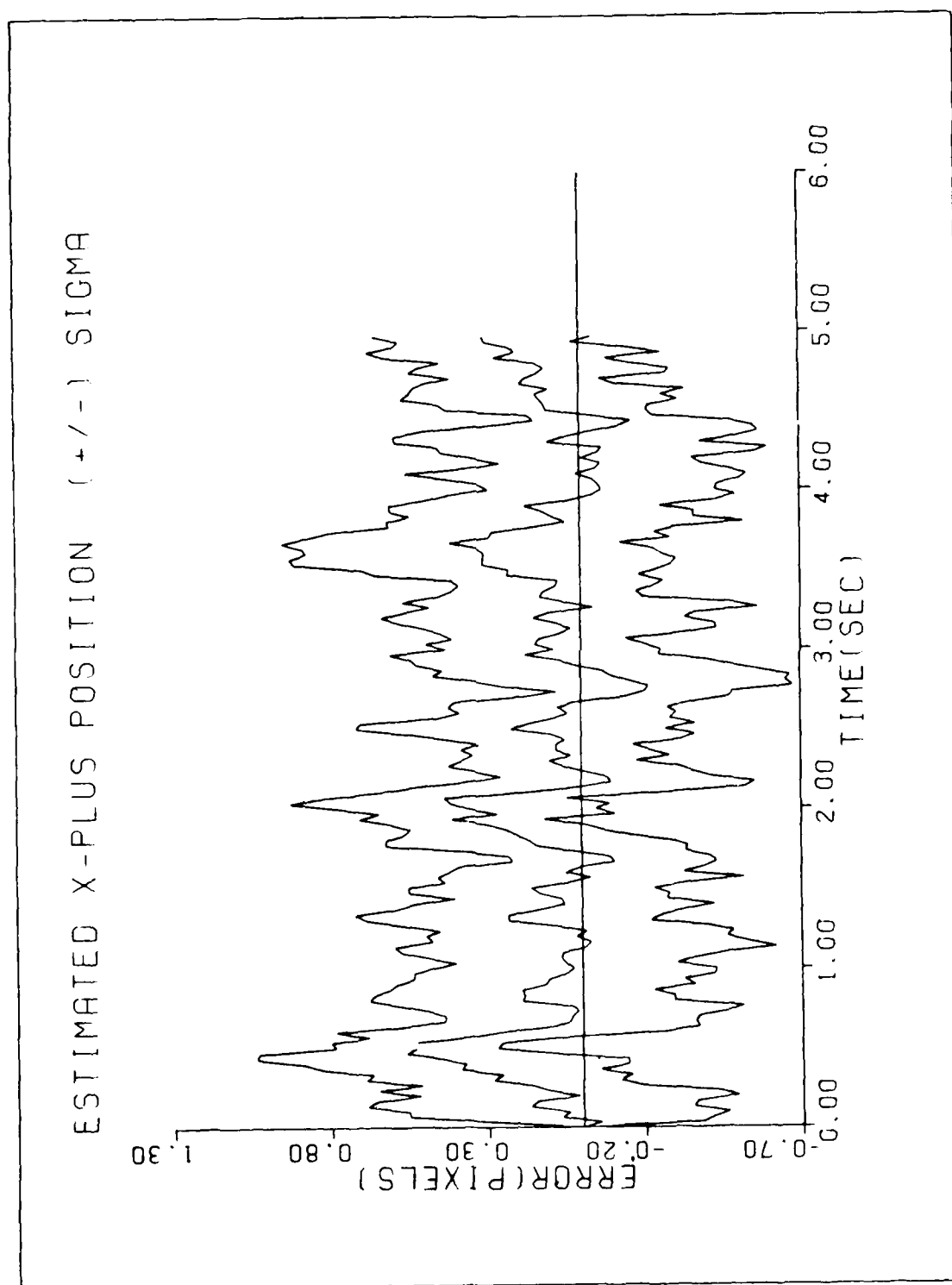


Figure F-5. $GM/MIAF/T2/2-G/T=8,3/KP=2$ nrad

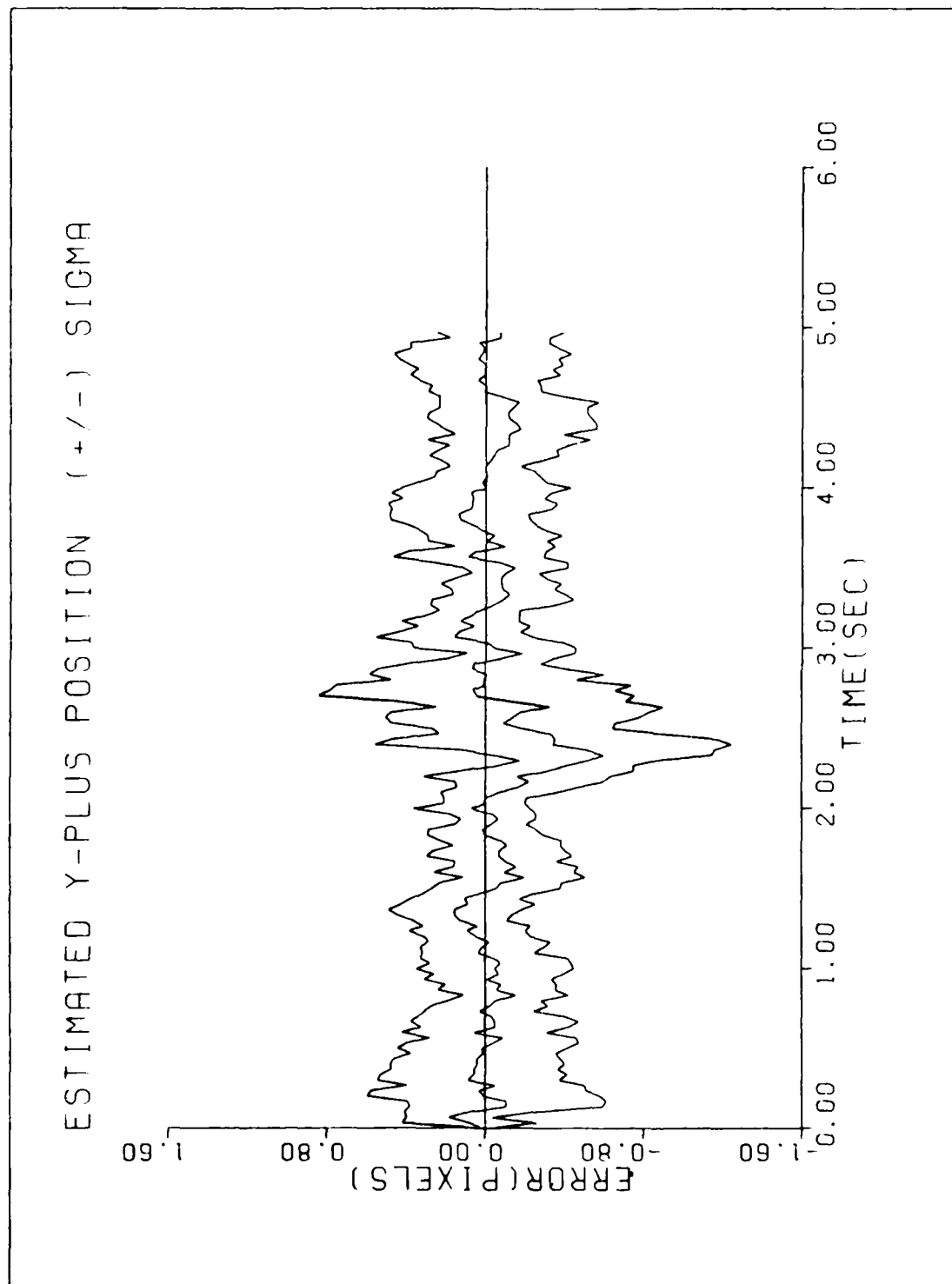


Figure F-6. GH/MMAF/T2/2-G/ $\tau=8,3$ /KP=2 nrad

ESTIMATED X-PLUS CENTROID POSITION (+/-) SIGMA

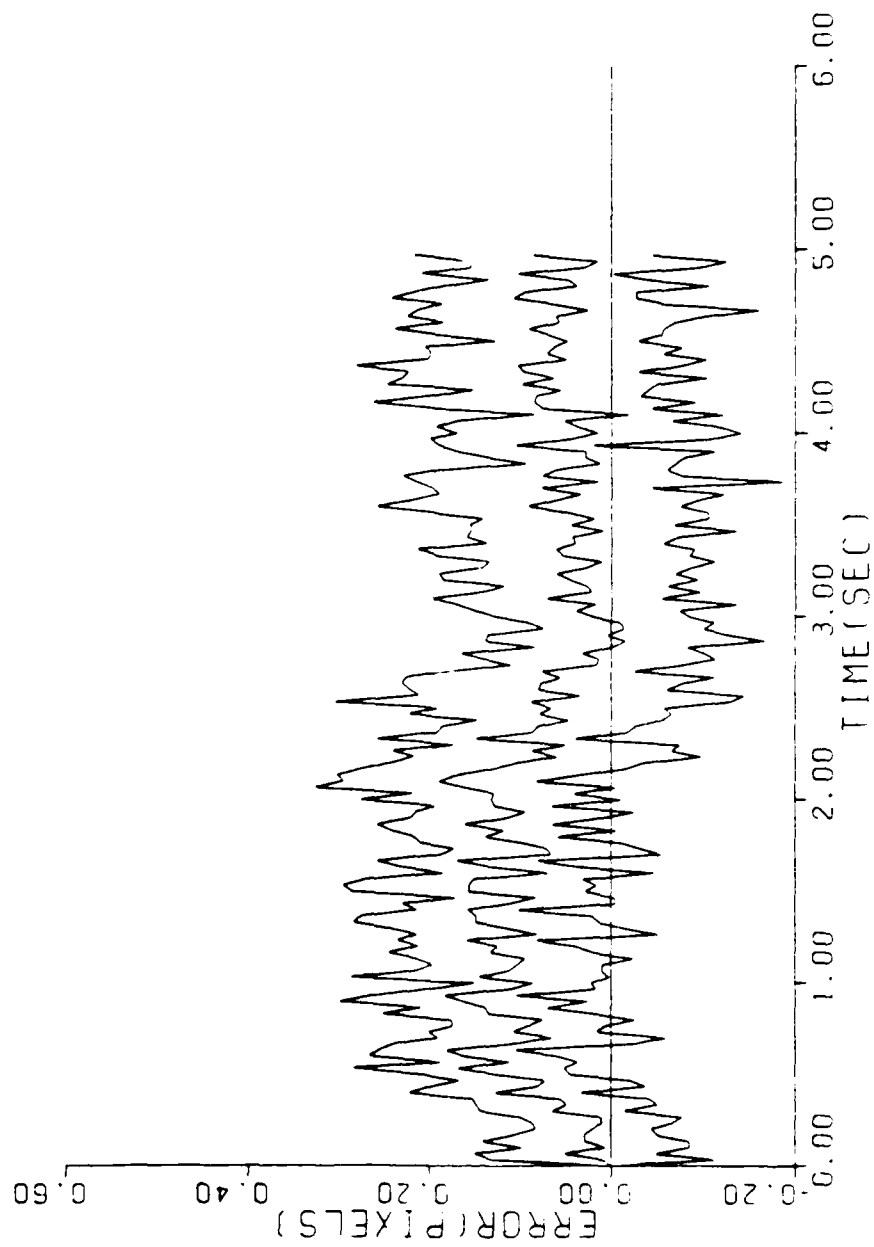


Figure F-7. $GM/MMAF/T2/2-G/\tau=8,3/KP=2$ nrad

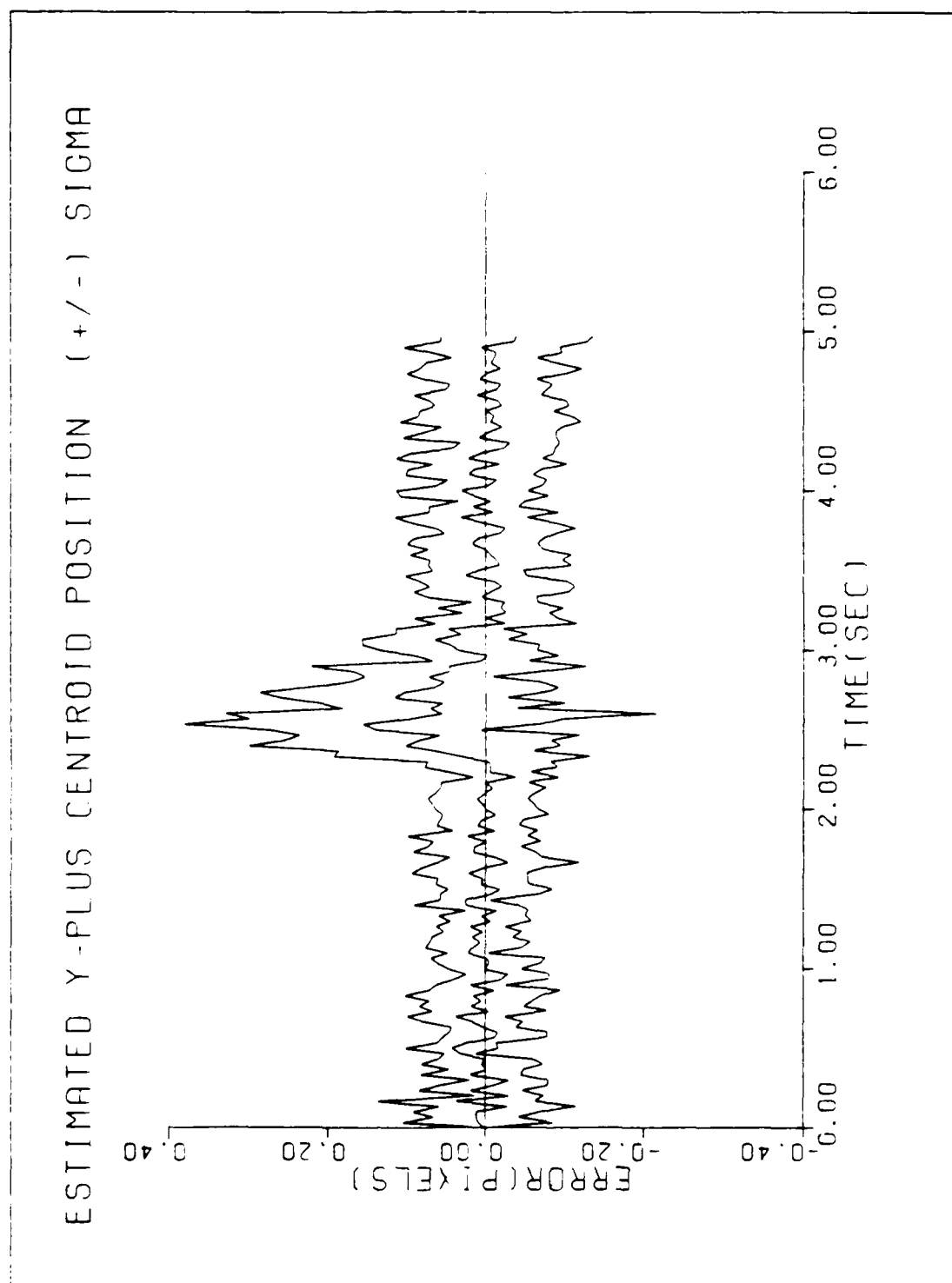


Figure F-8. $GM/MIAF/T2/2-G/T=8,3/KP=2$ nrad

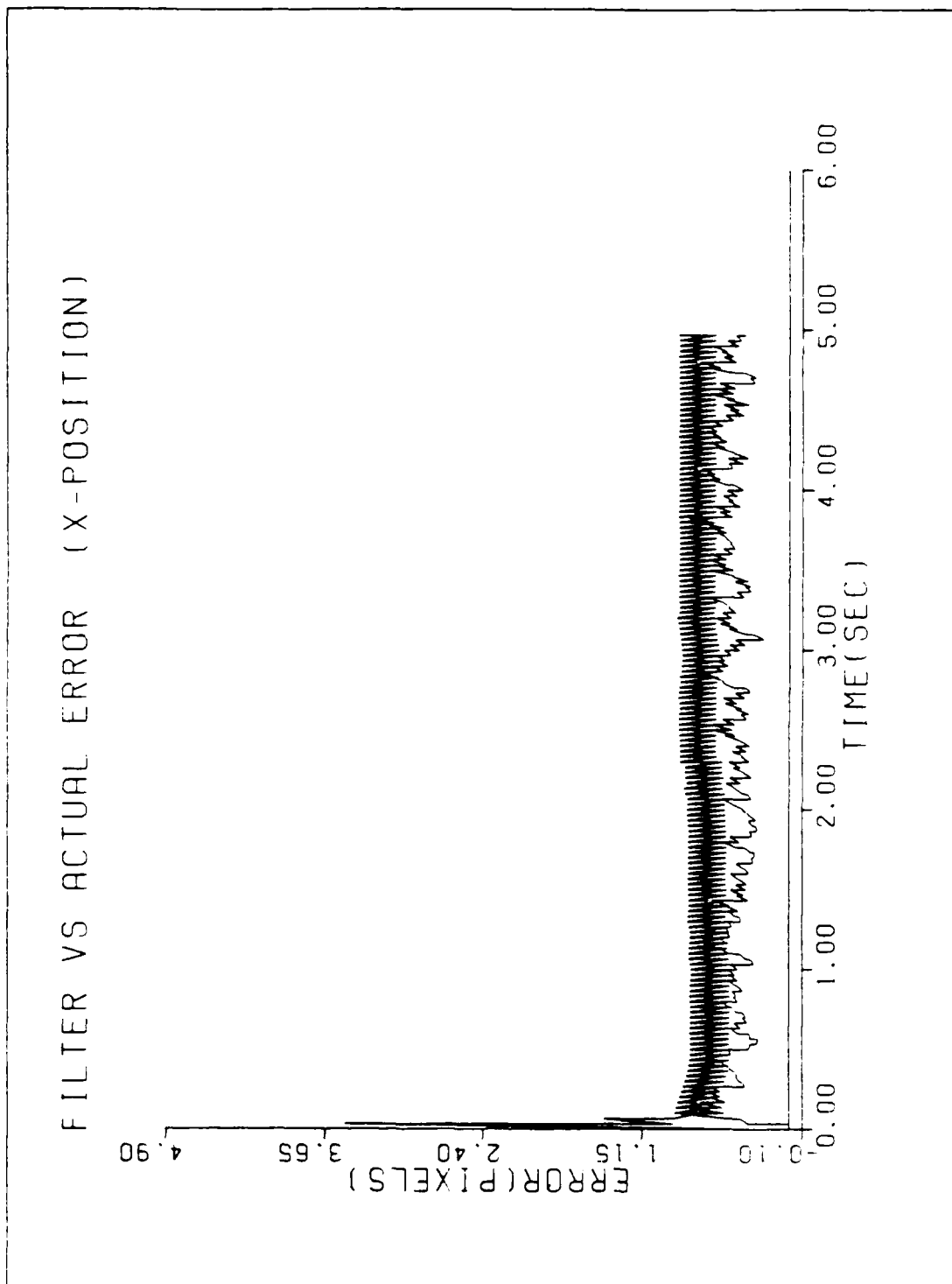


Figure F-9. GM/MMAF/T2/20-G

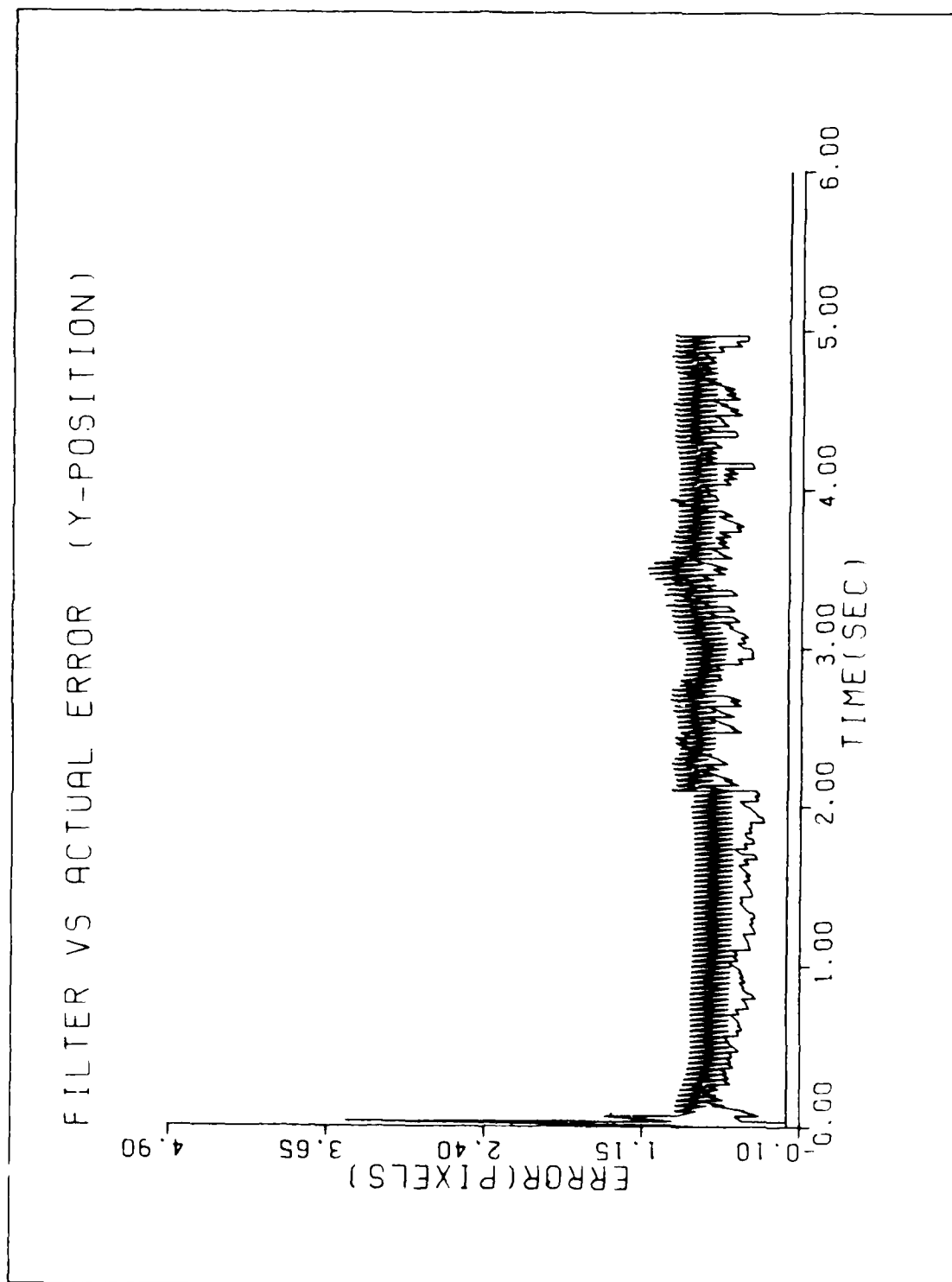


Figure F-10. GM/MMAF/T2/20-G

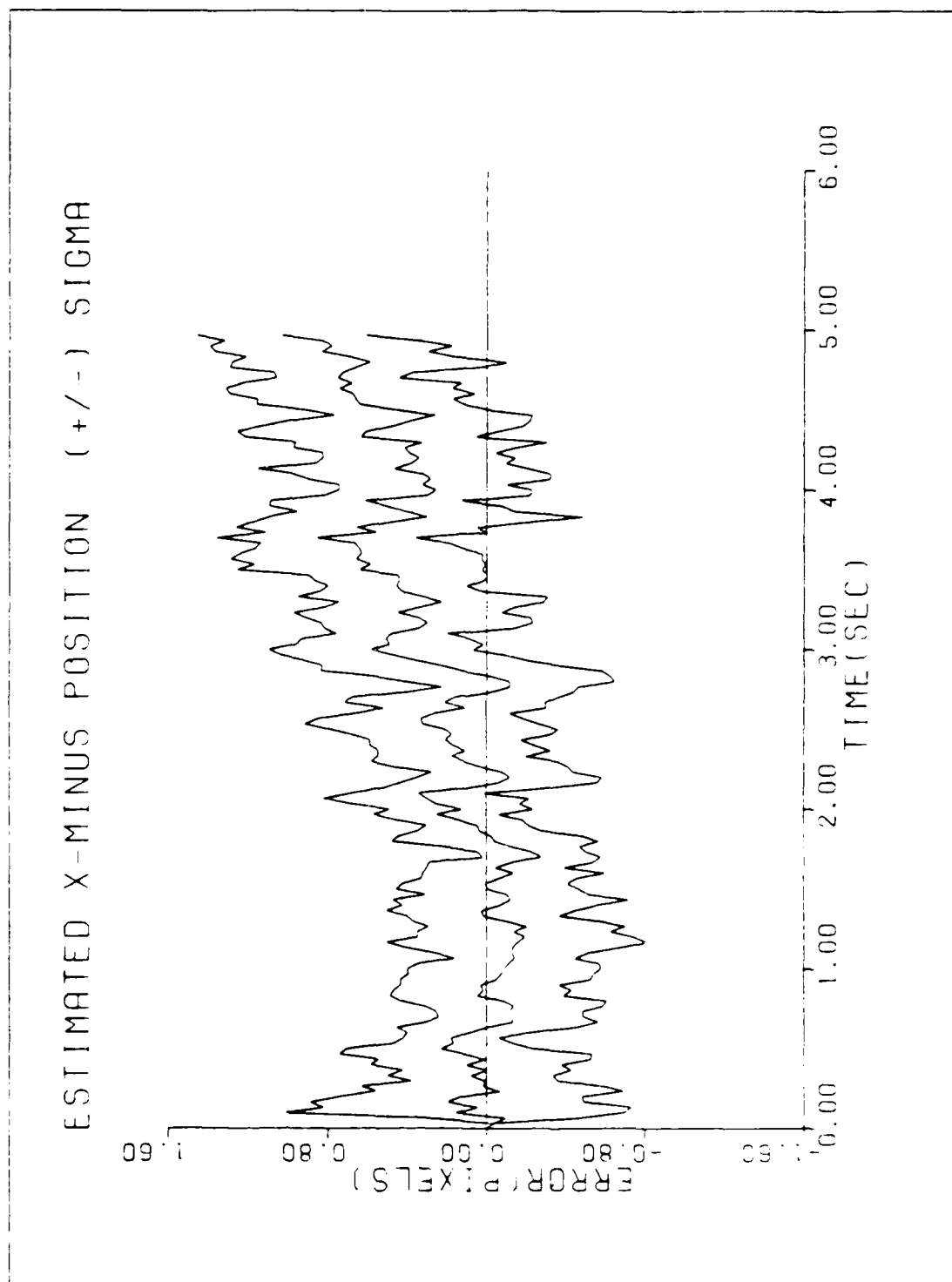


Figure F-11. GM/MAF/T2/20-G

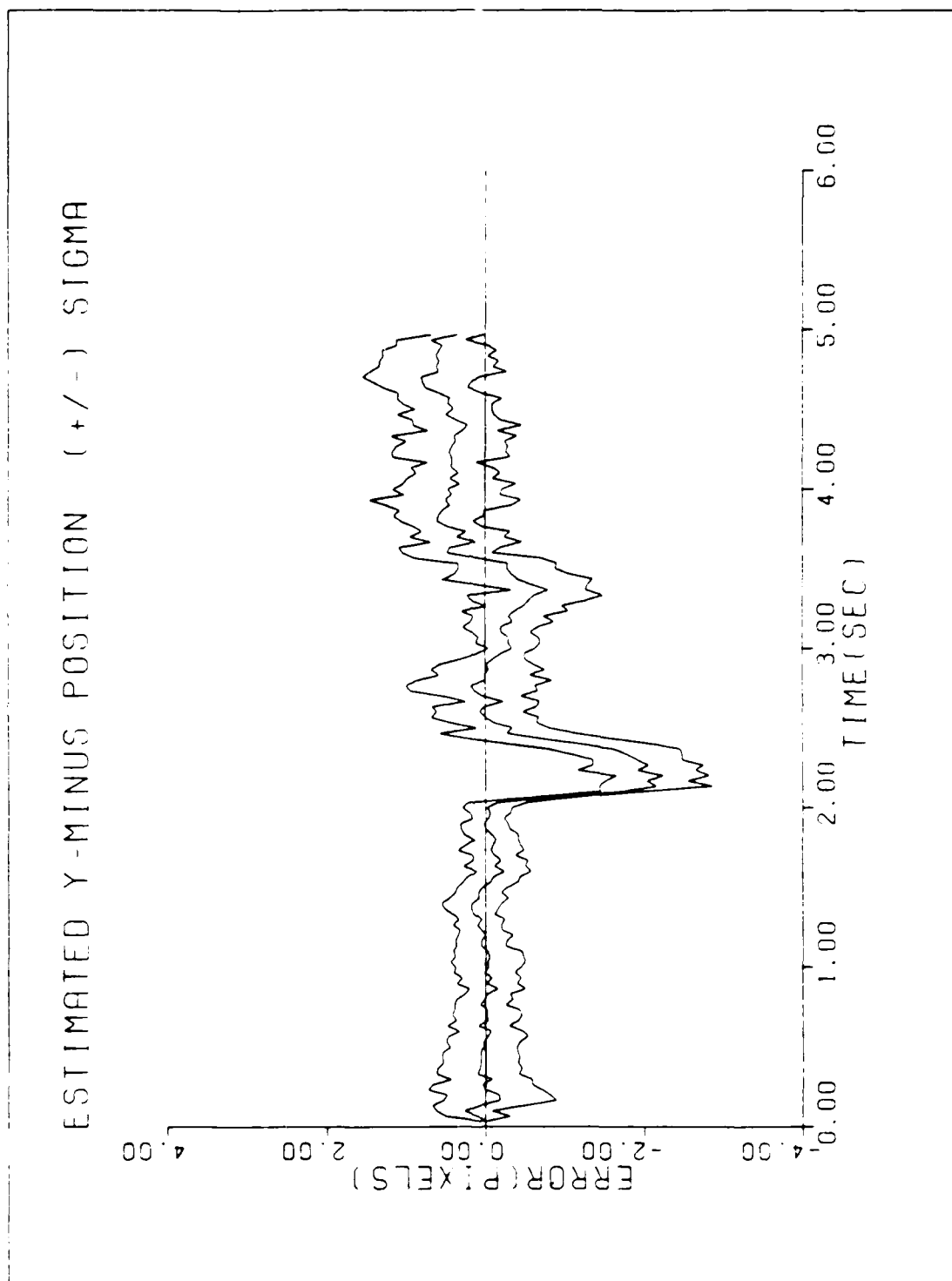


Figure F-12. GM/MMAF/T2/20-G

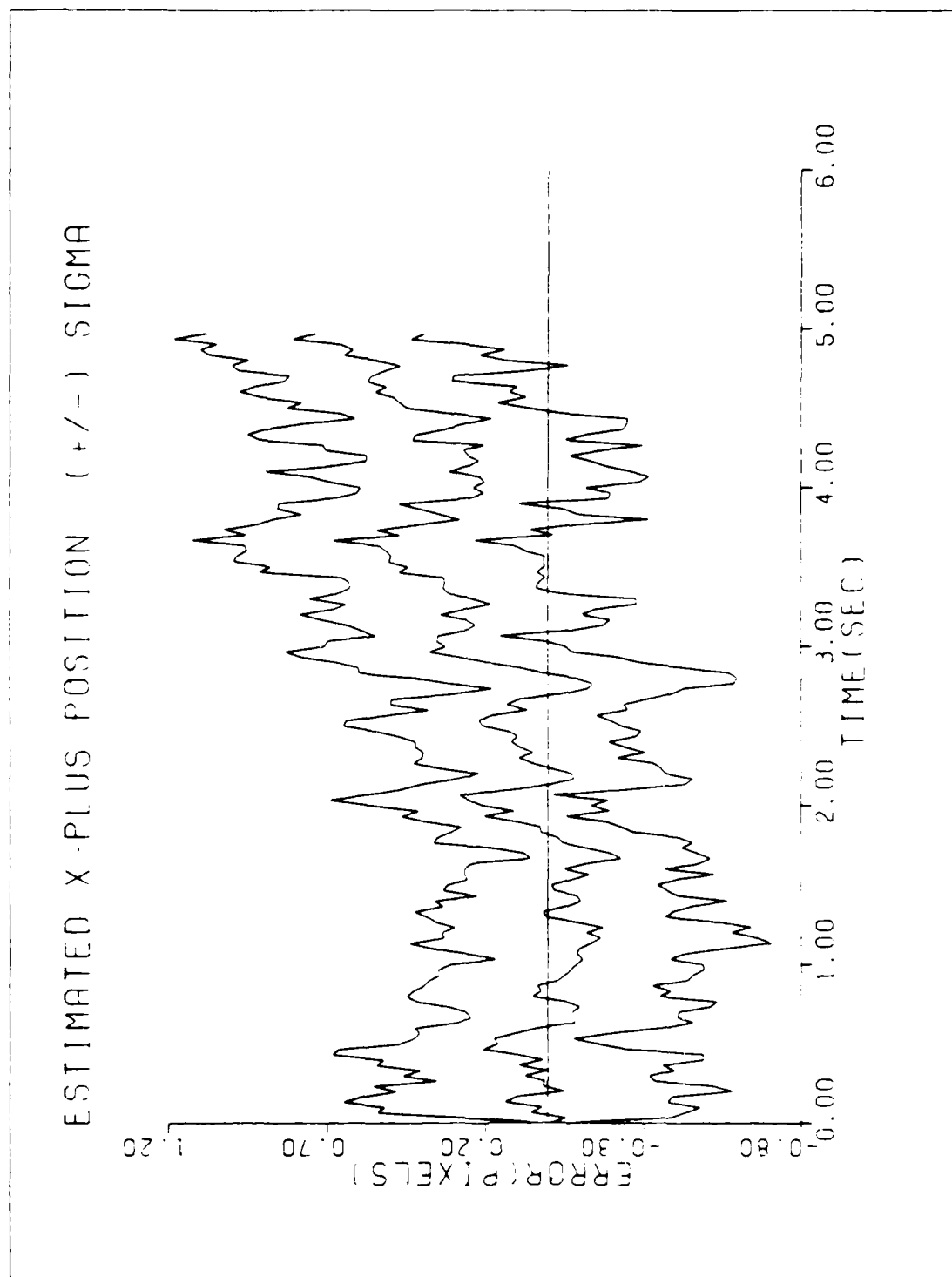


Figure F-13. GM/MMAF/T2/20-G

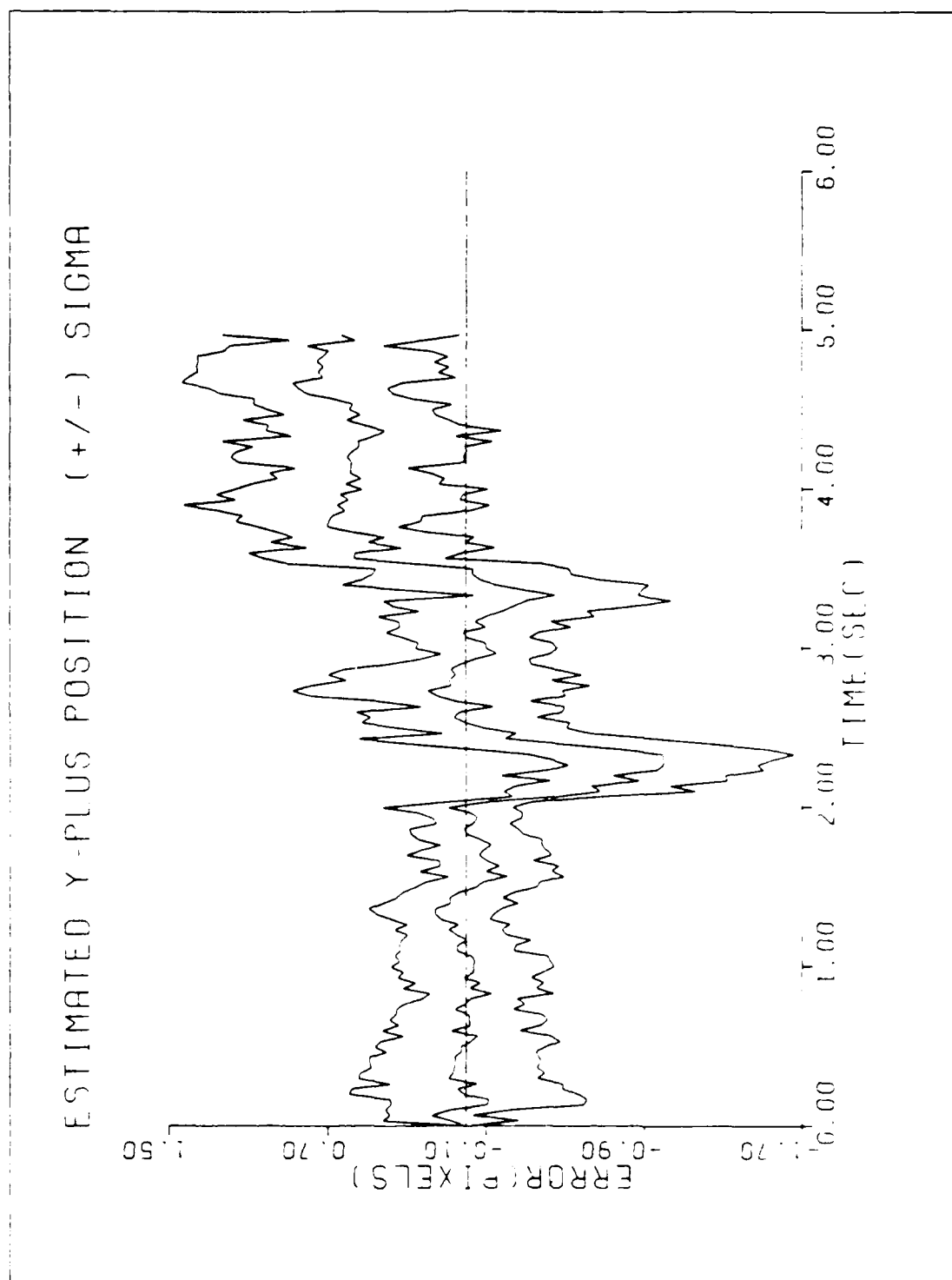


Figure F-14. GM/MMAF/T2/20-G

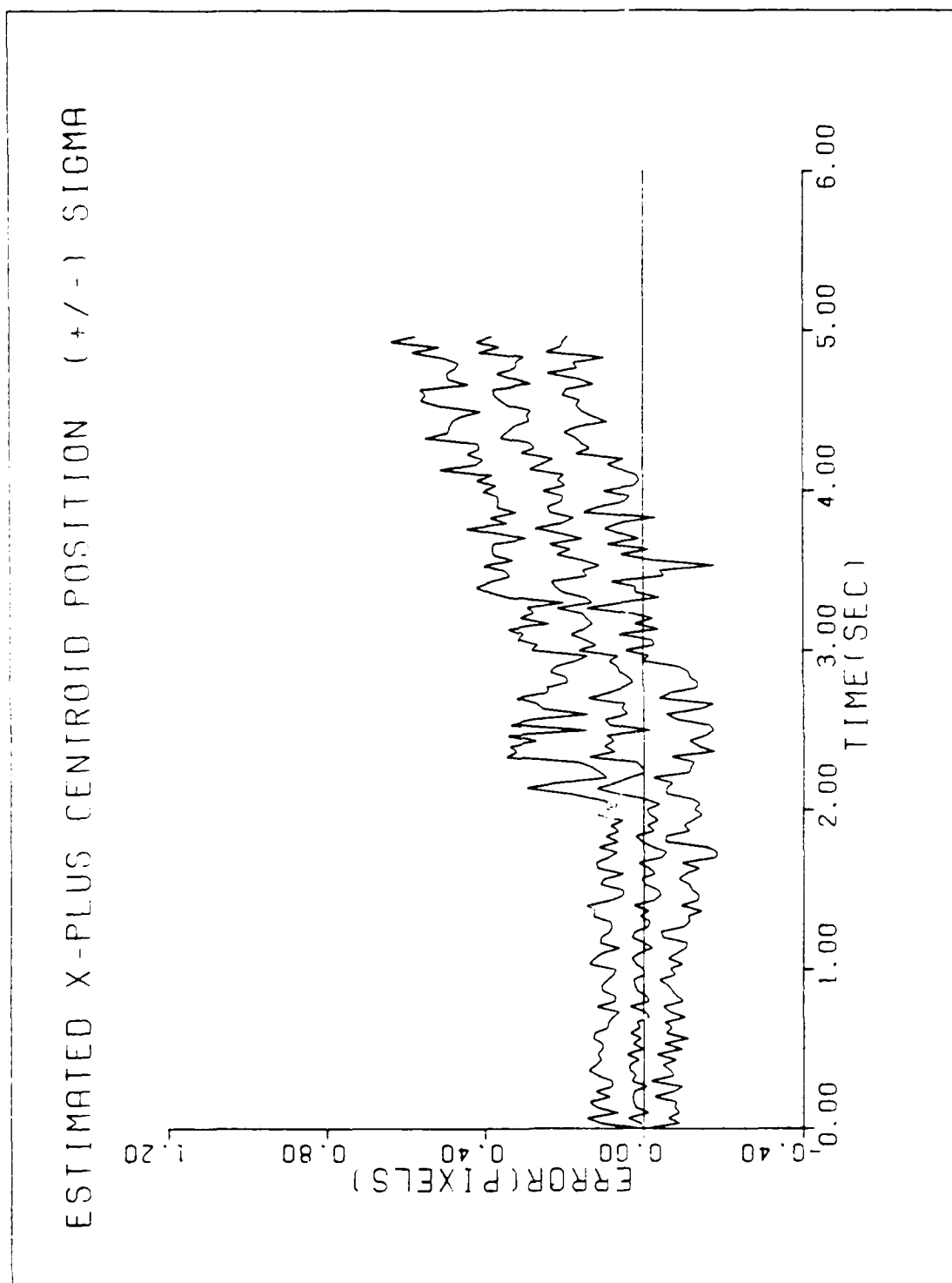


Figure F-15. GM/MMAF/T2/20-6

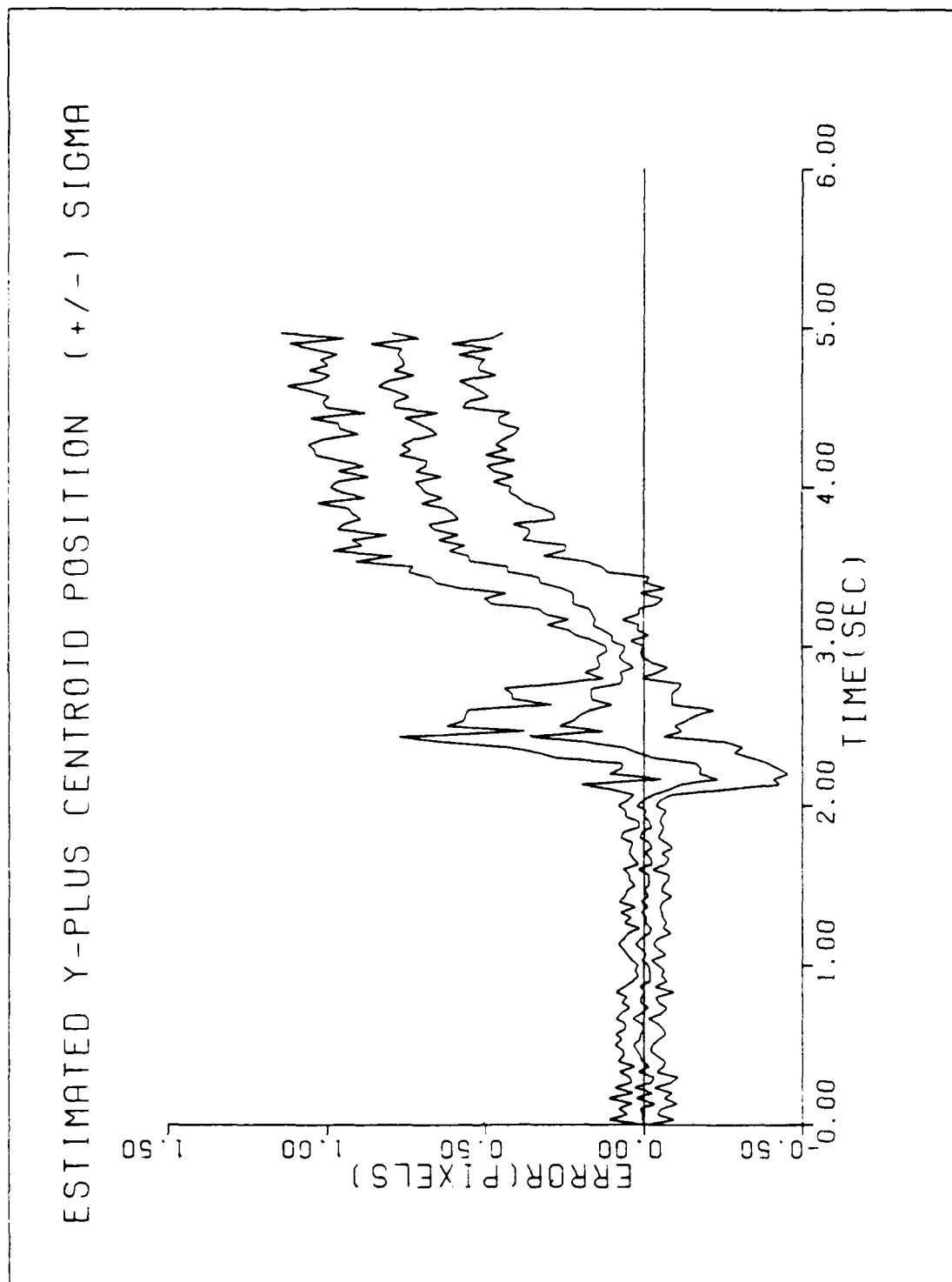


Figure F-16. GM/MMAF/T2/20-G

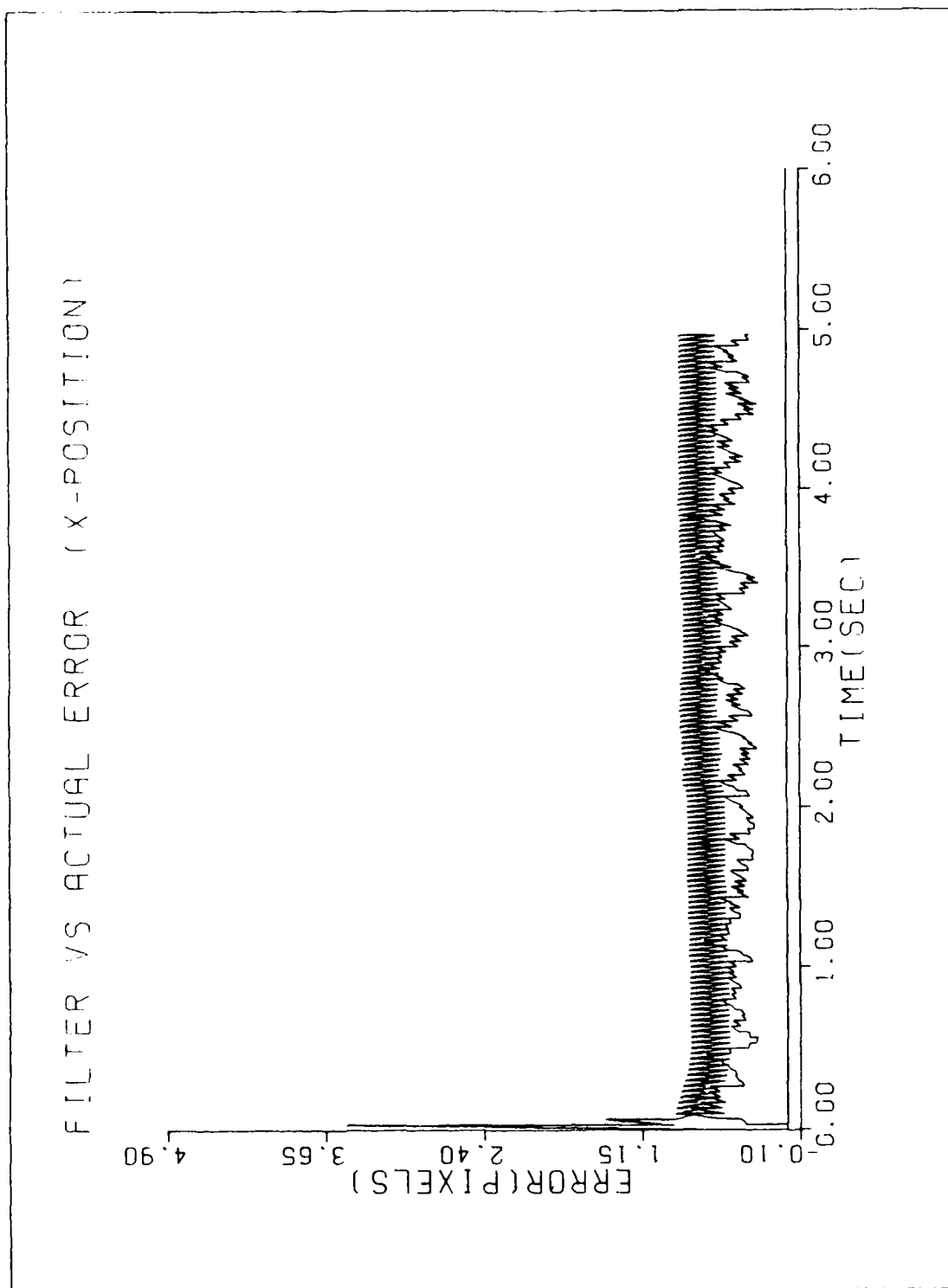


Figure F-17. GM/MAF/T2/10-G/ $\tau=4,.8$

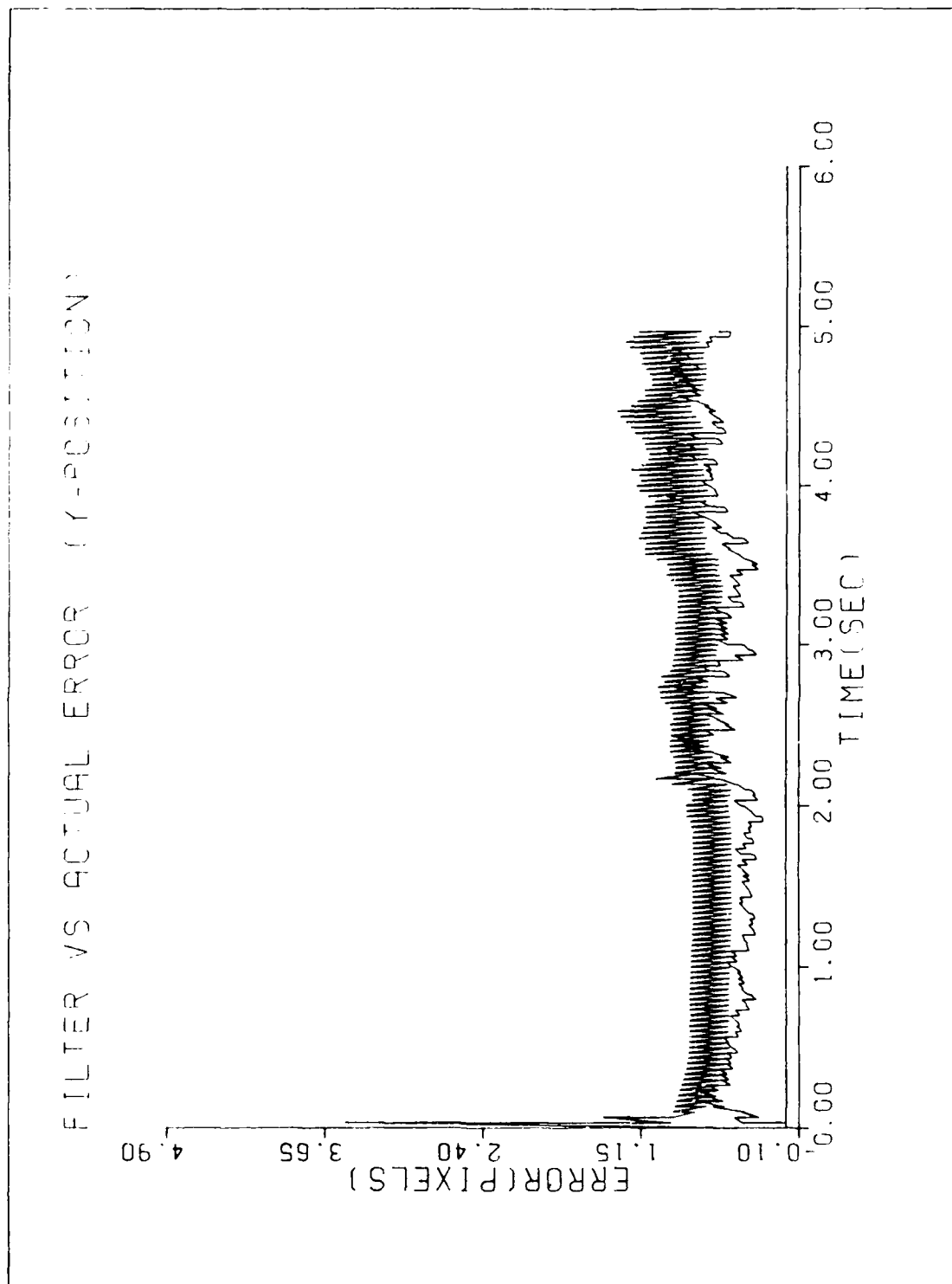


Figure F-18. GM/MMAF/T2/10-G/ $\tau=4,.8$

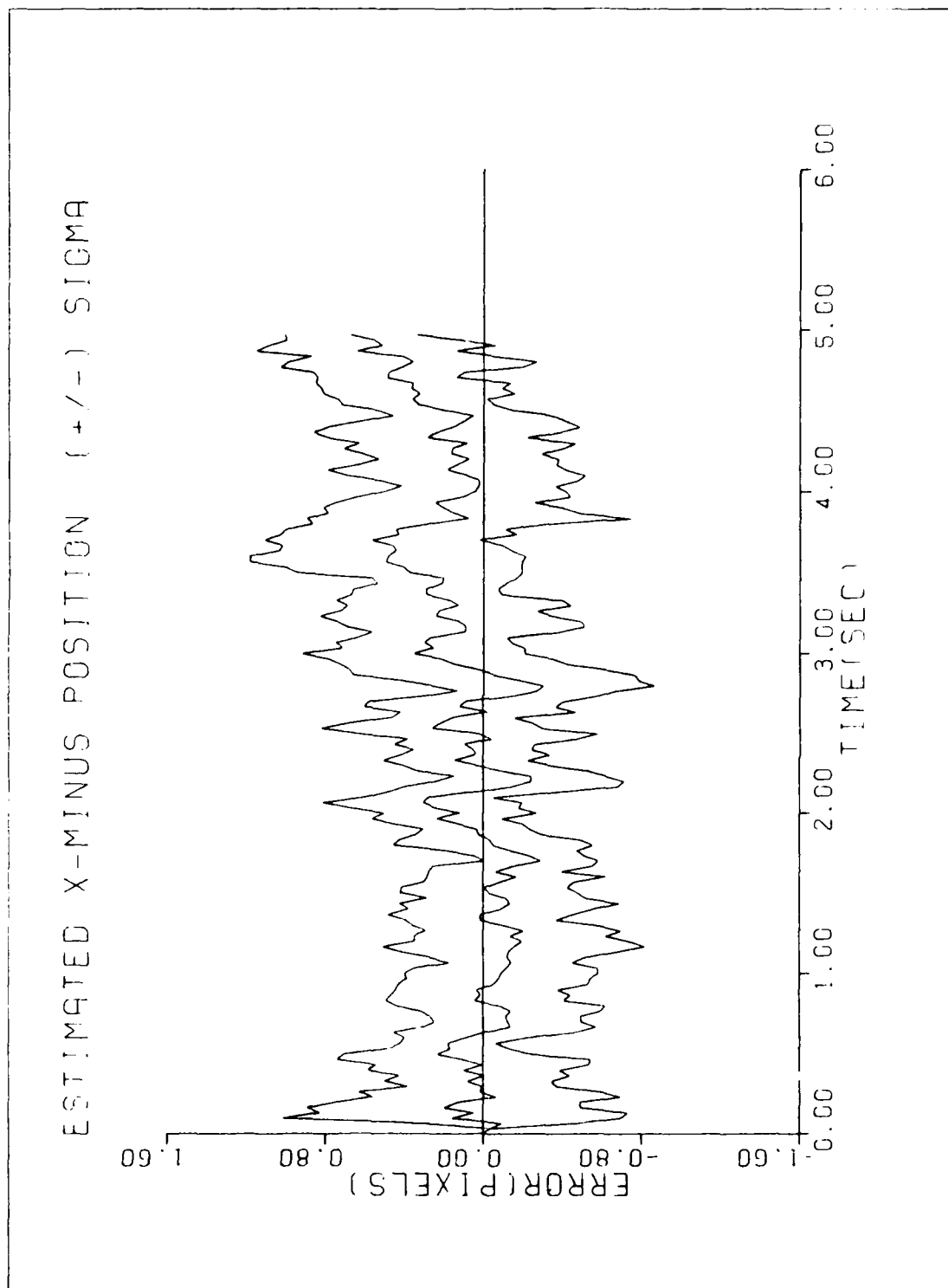


Figure F-19. $GH/MMAF/T2/10-G/T=4,.8$

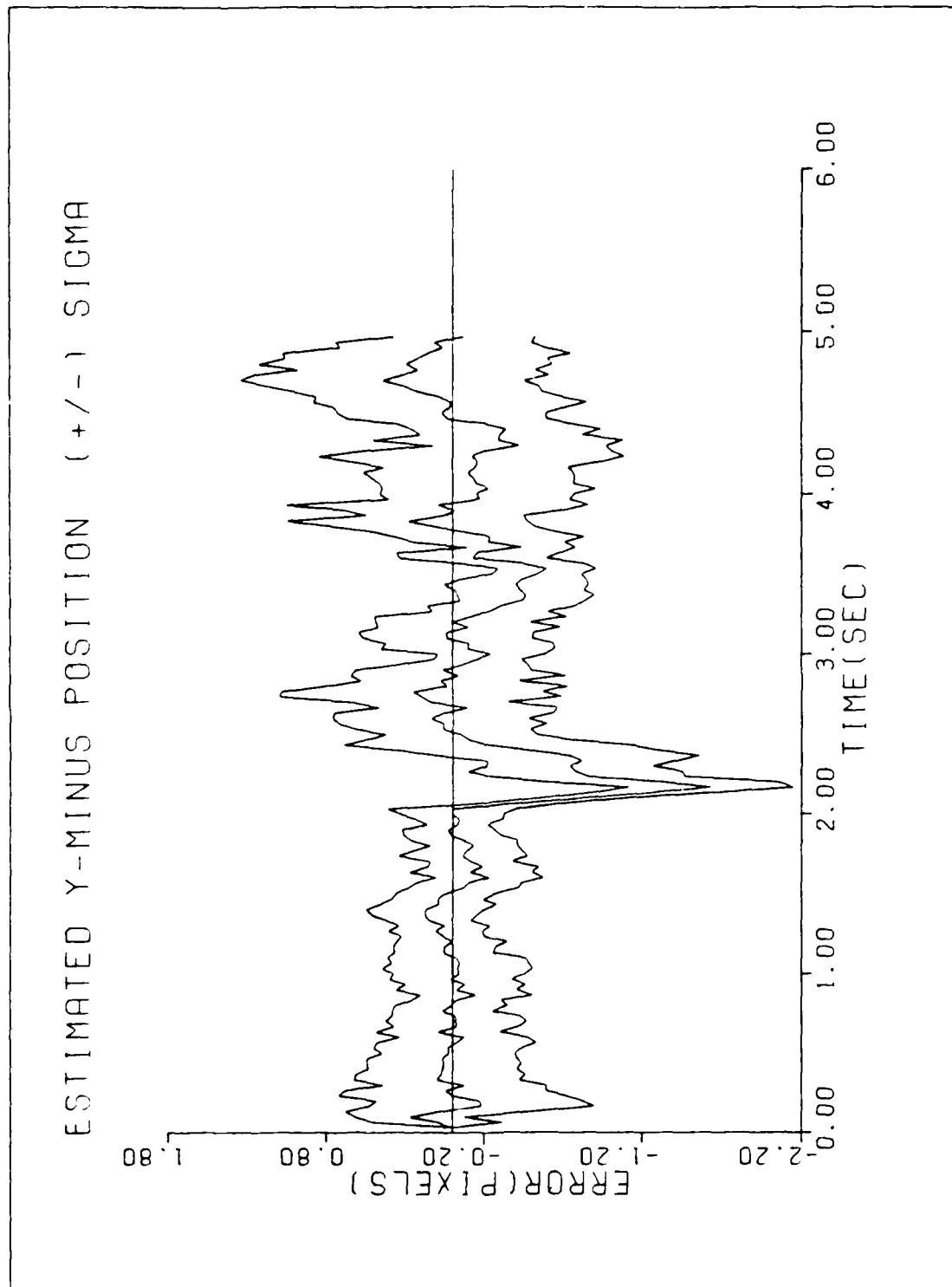


Figure F-20. $GM/MMAF/T2/10-G/\tau=4.8$

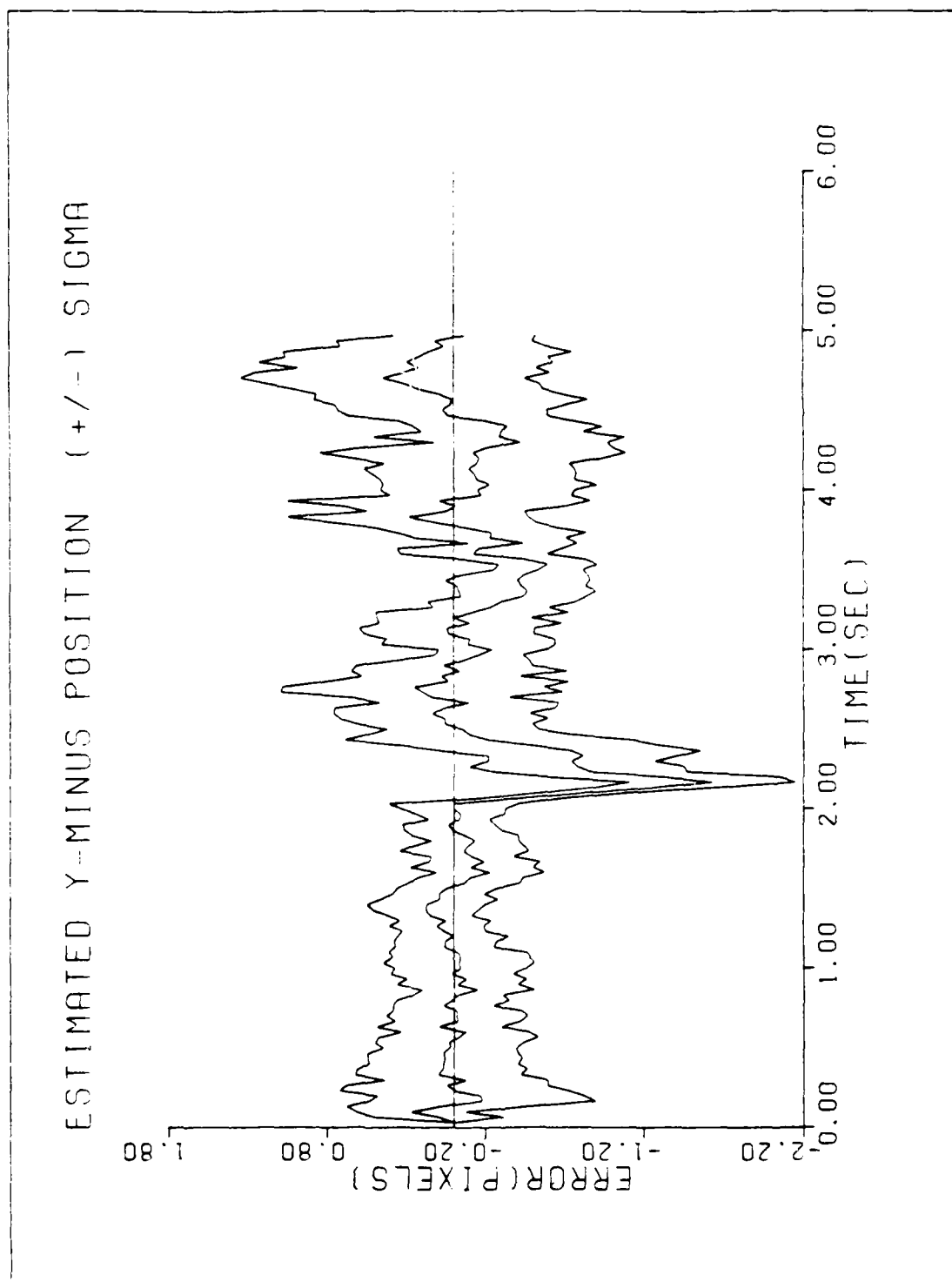


Figure F-20. $GM/MMAF/T2/10-G/\tau=4,.8$

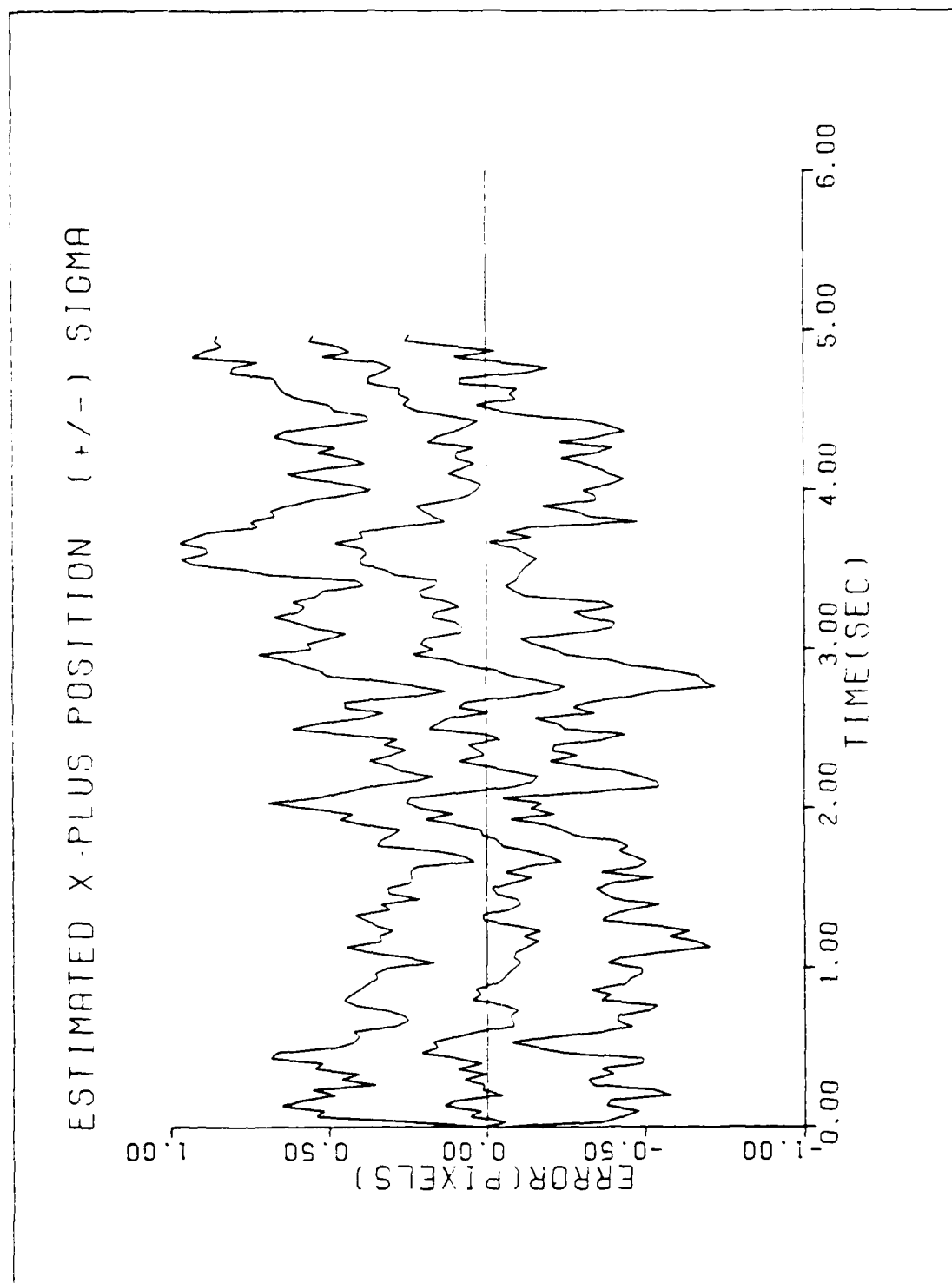


Figure F-21. GM/MMAF/T2/10-G/ $\tau=4.8$

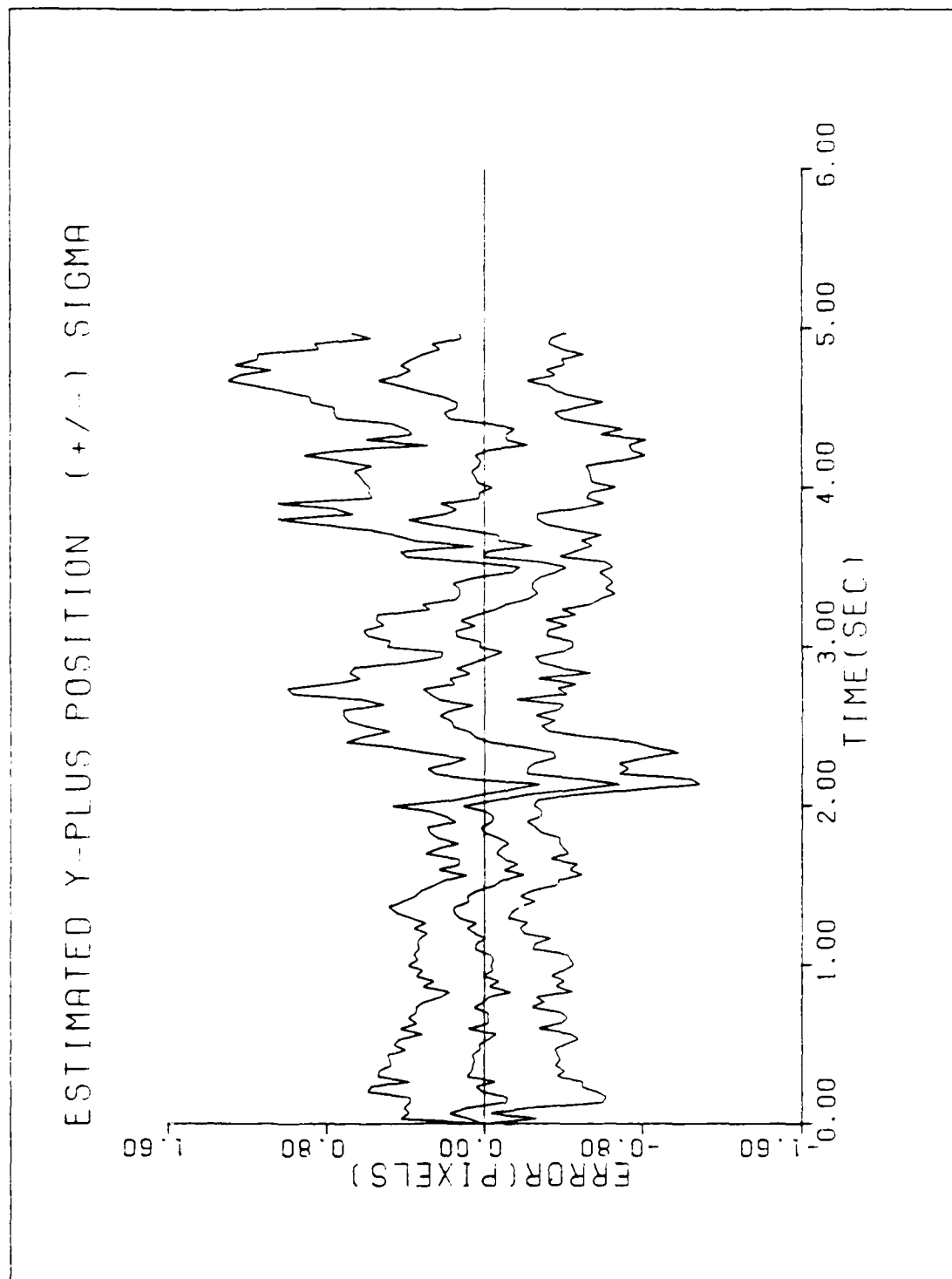


Figure F-22. GM/MMAF/T2/10-G/ $\tau=4,.8$

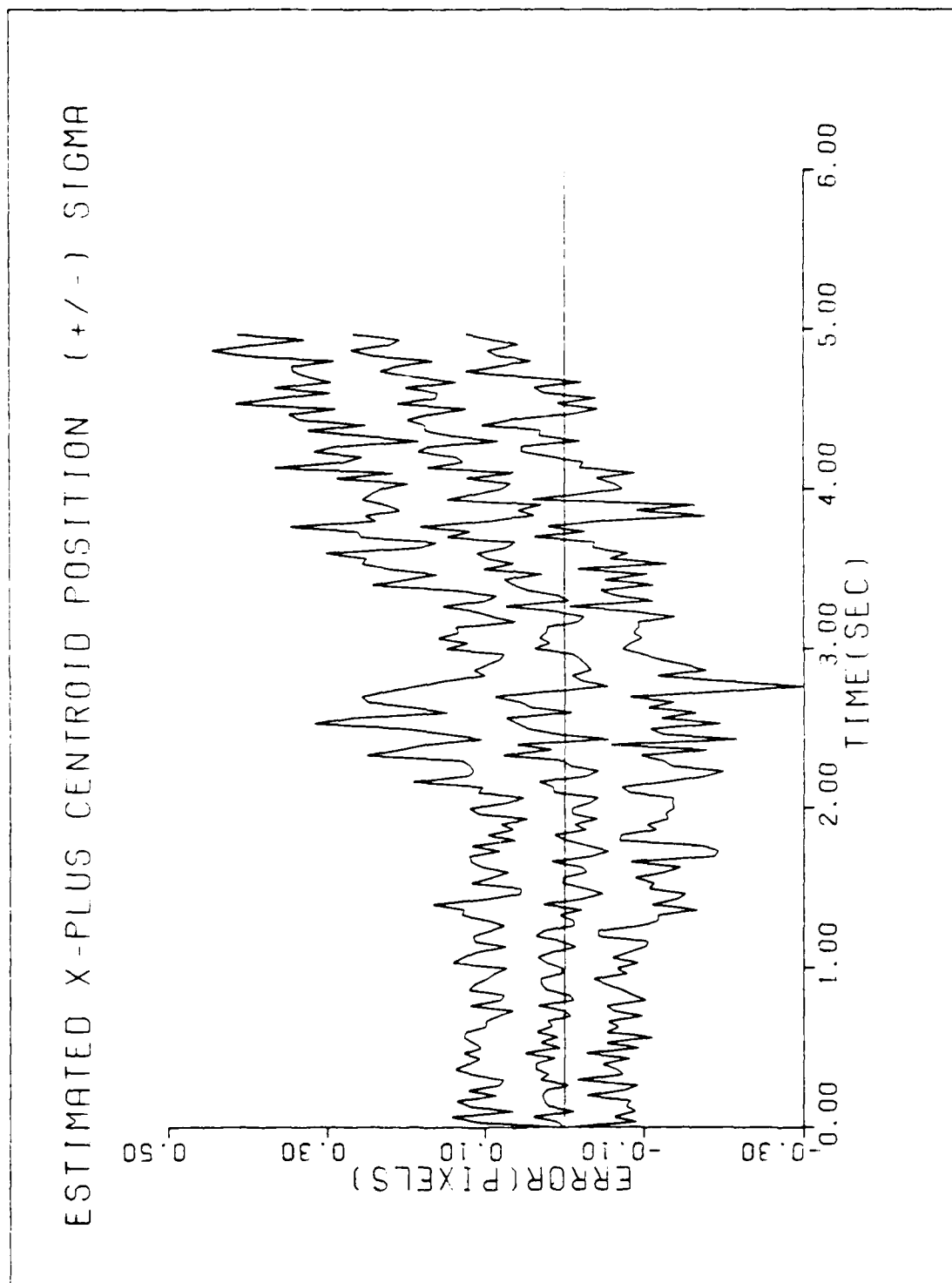


Figure F-23. GM/MMAF/T2/10-6/ $\tau=4,.8$

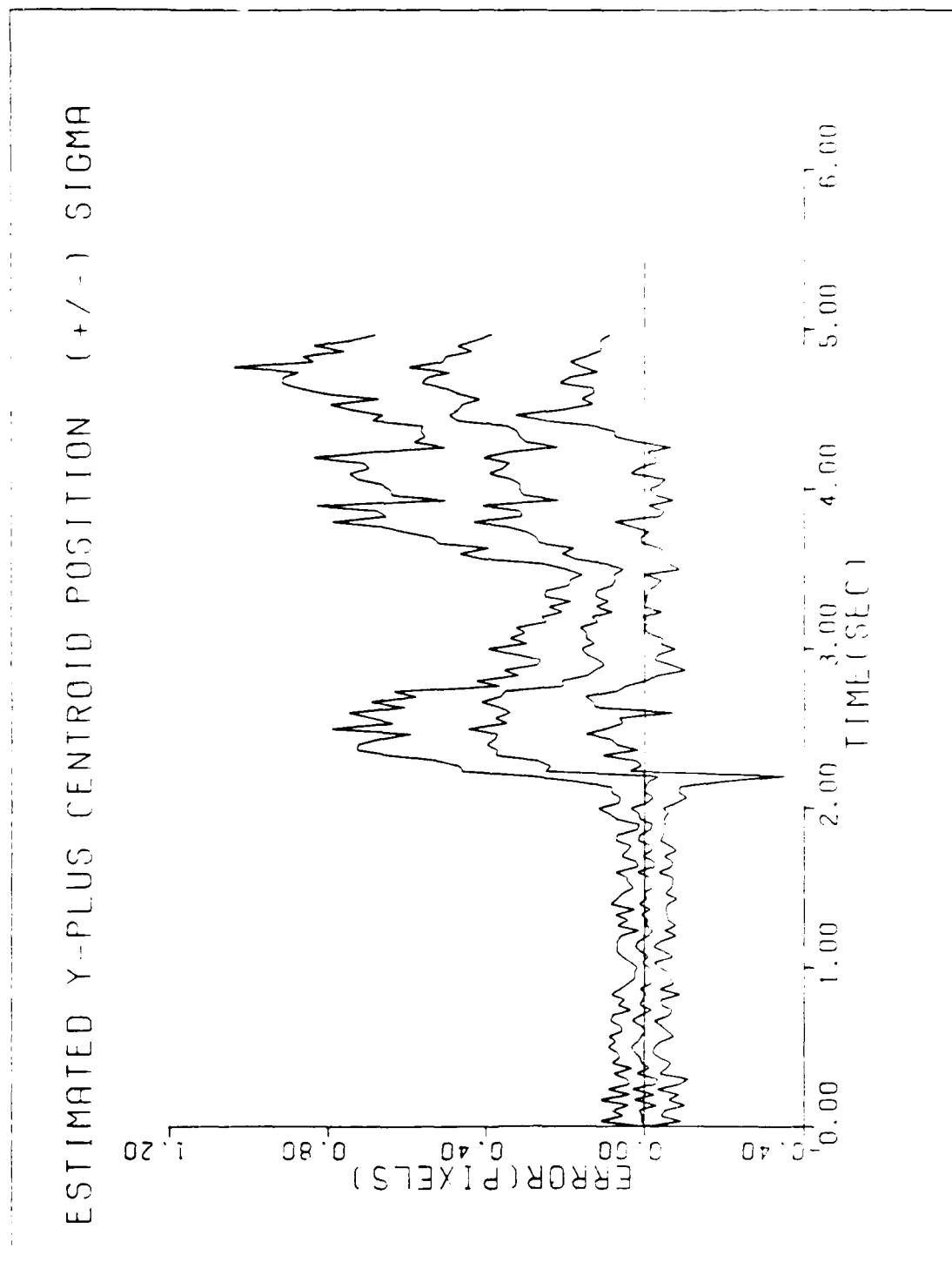


Figure F-24. GM/MMAF/T2/10-G/ $\tau=4,.8$

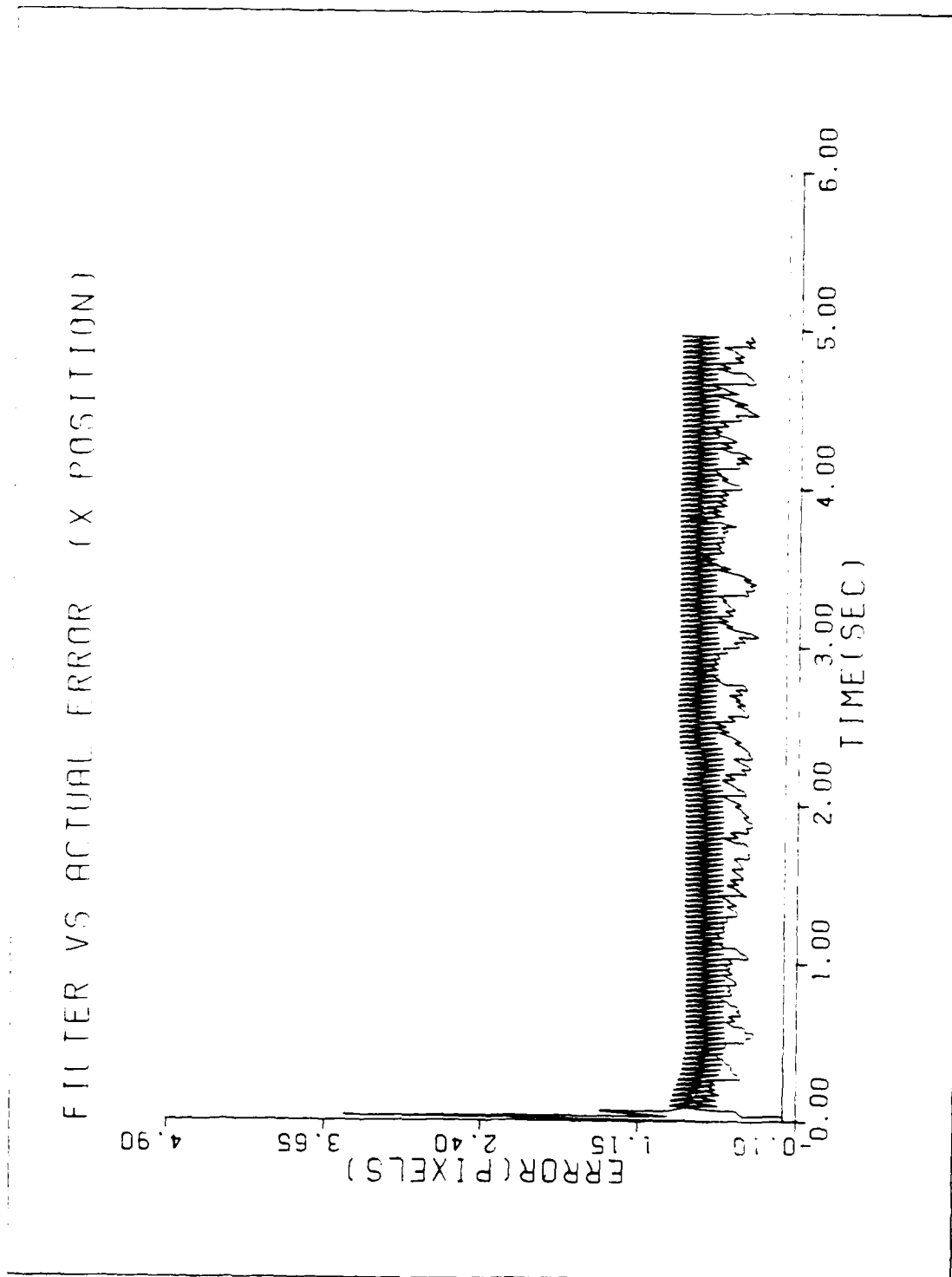


Figure F-25. GM/MMAF/T2/20-G/ $\tau=4,.4$

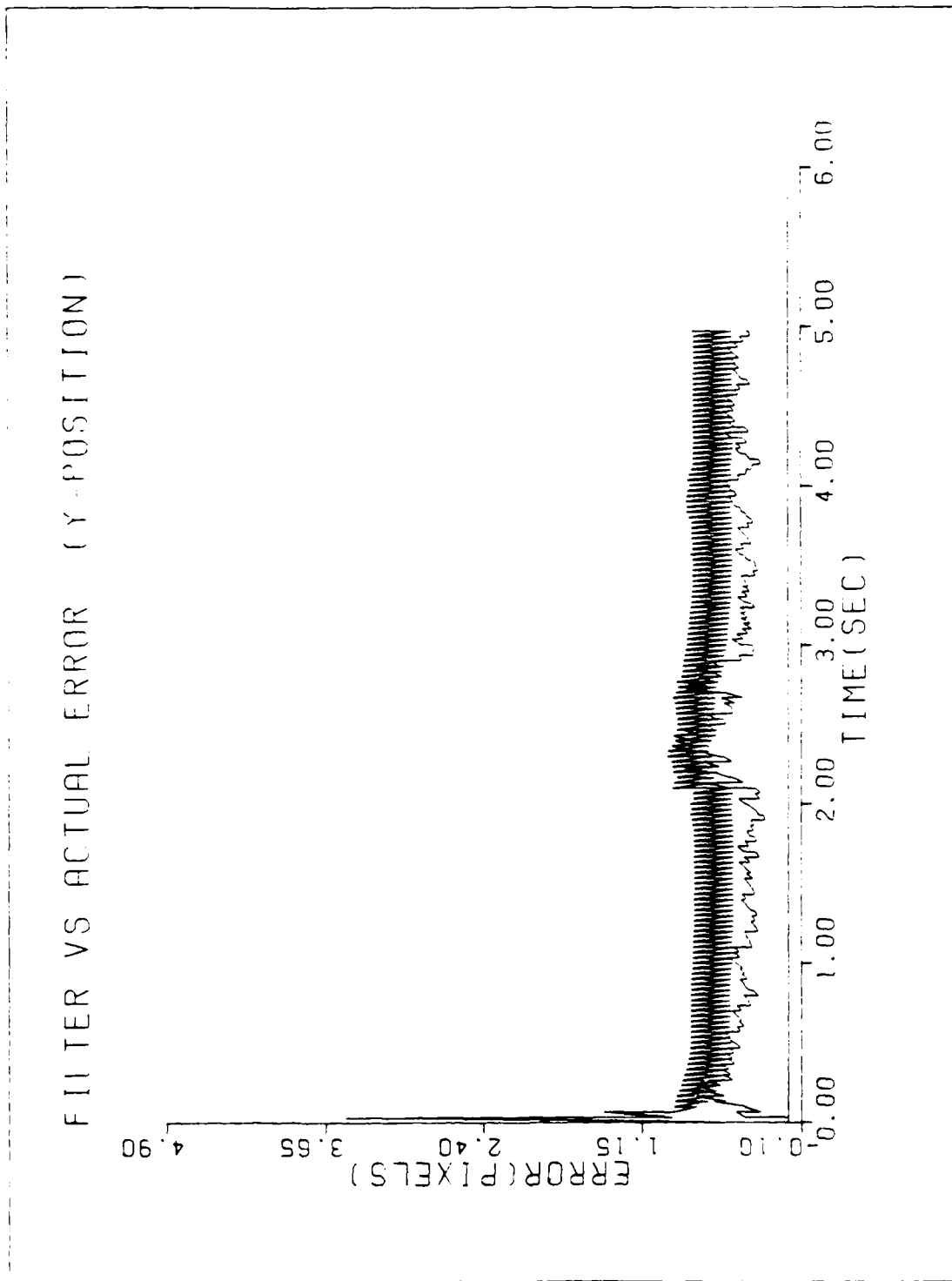


Figure F-26. GM/MMAF/T2/20-G/ $\tau=4,.4$

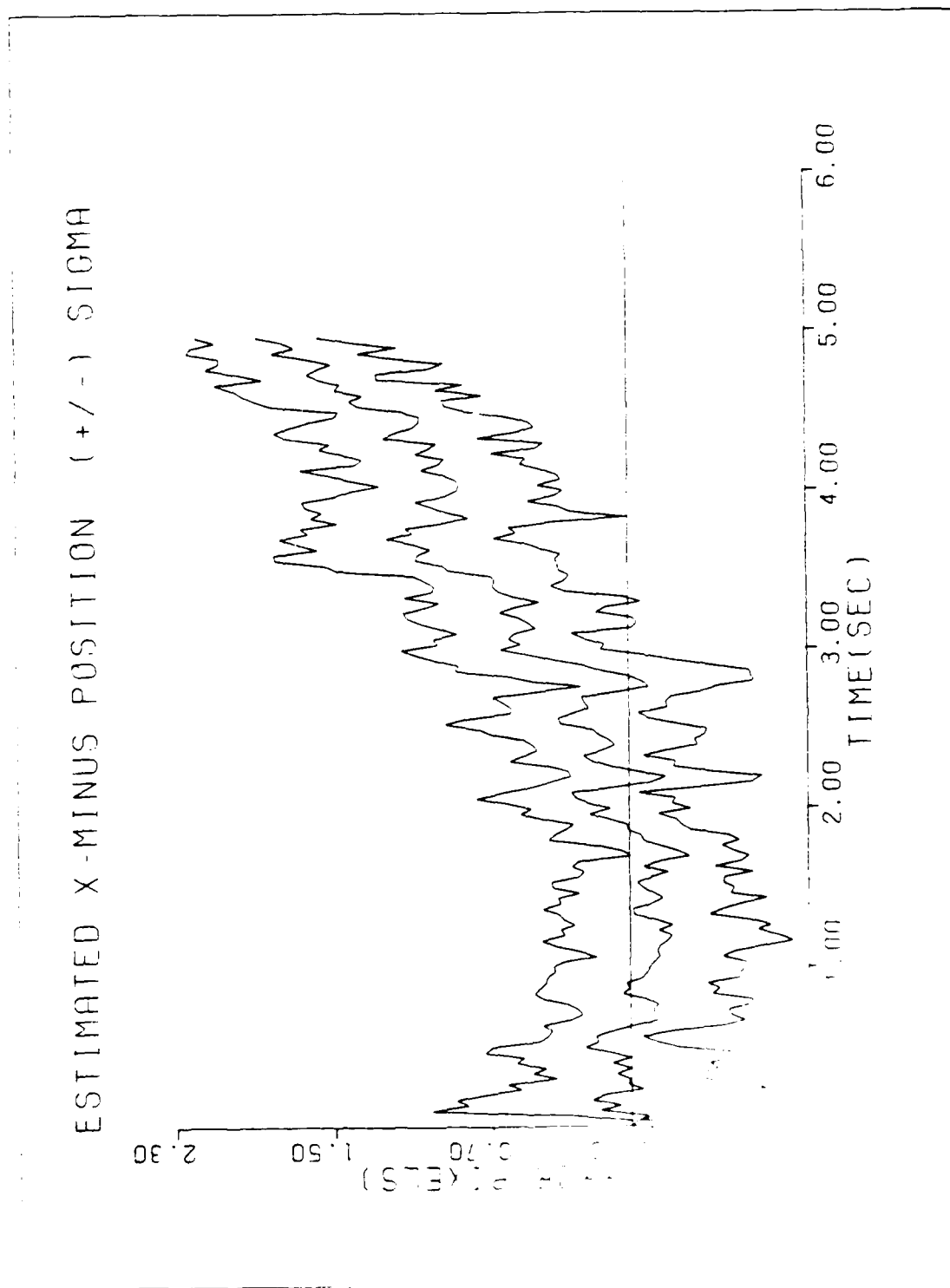


Figure F-27 GM 100A

AD-A190 619

A MULTIPLE MODEL ADAPTIVE TRACKING ALGORITHM AGAINST

4/4

AIRBORNE TARGETS(U) AIR FORCE INST OF TECH

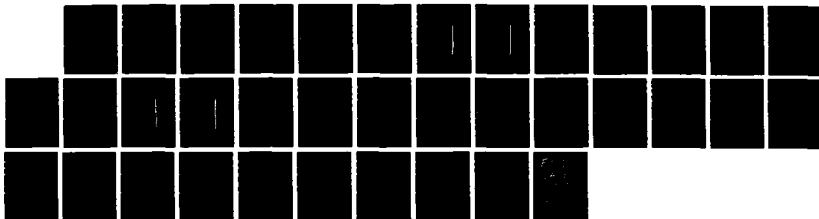
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING

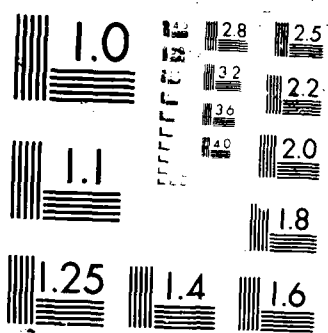
UNCLASSIFIED

T A LEENEY DEC 87 AFIT/OE/ENG/87D-37

F/G 17/5.1

ML





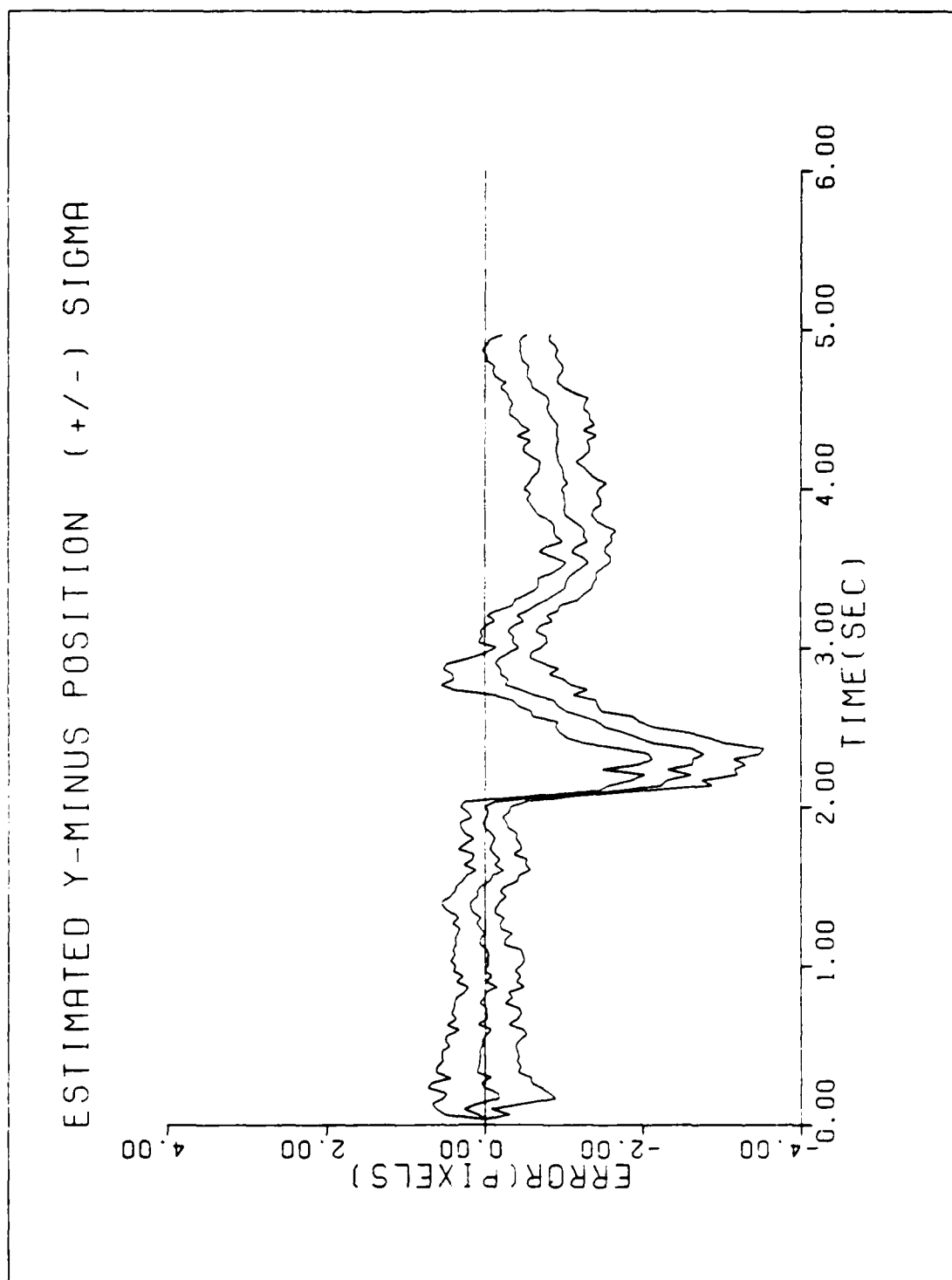


Figure F-28. $GM/MMAF/T2/20-G/\tau=4,.4$

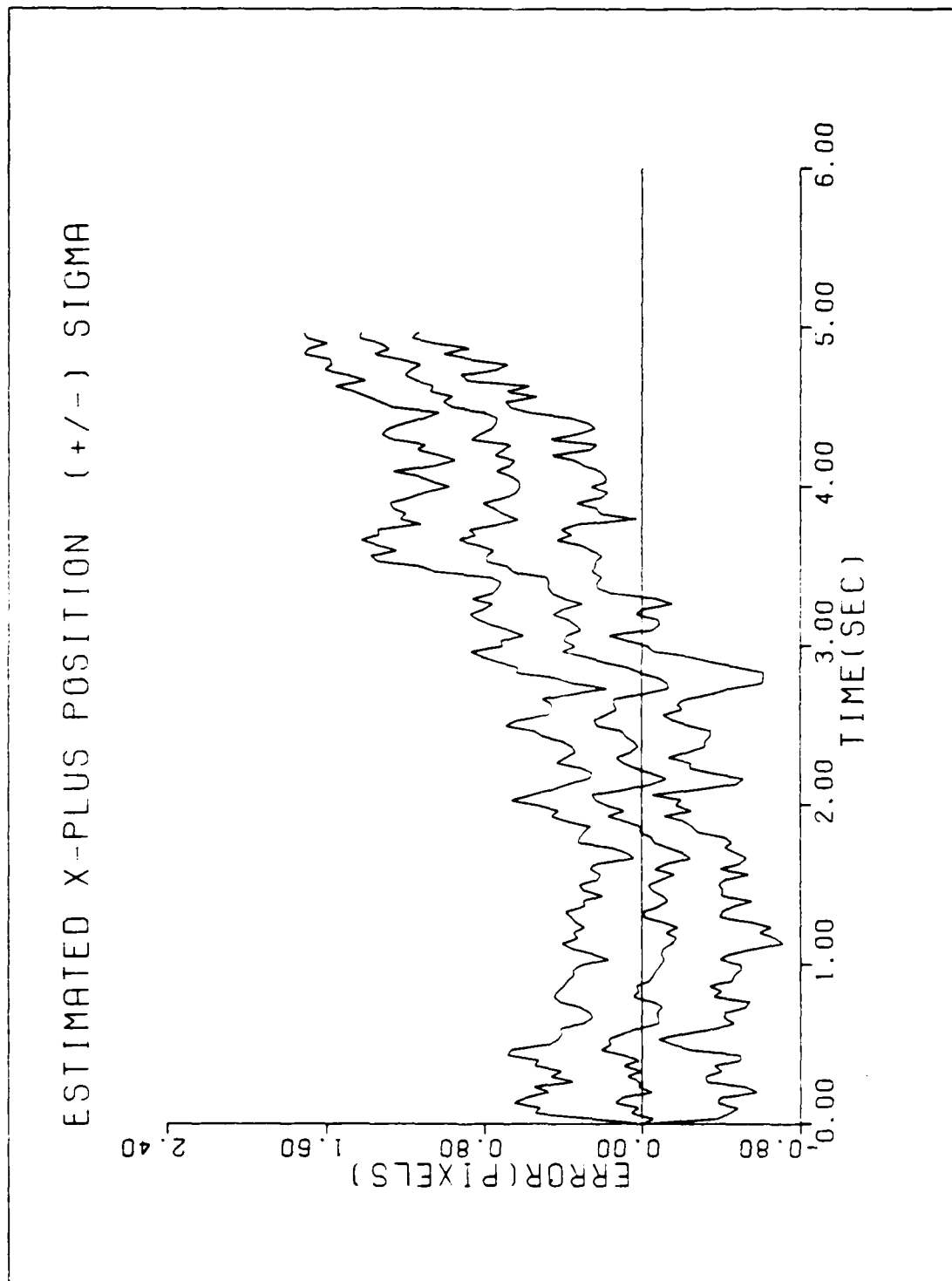


Figure F-29. GM/MMAF/T2/20-G/ $\tau=4, .4$

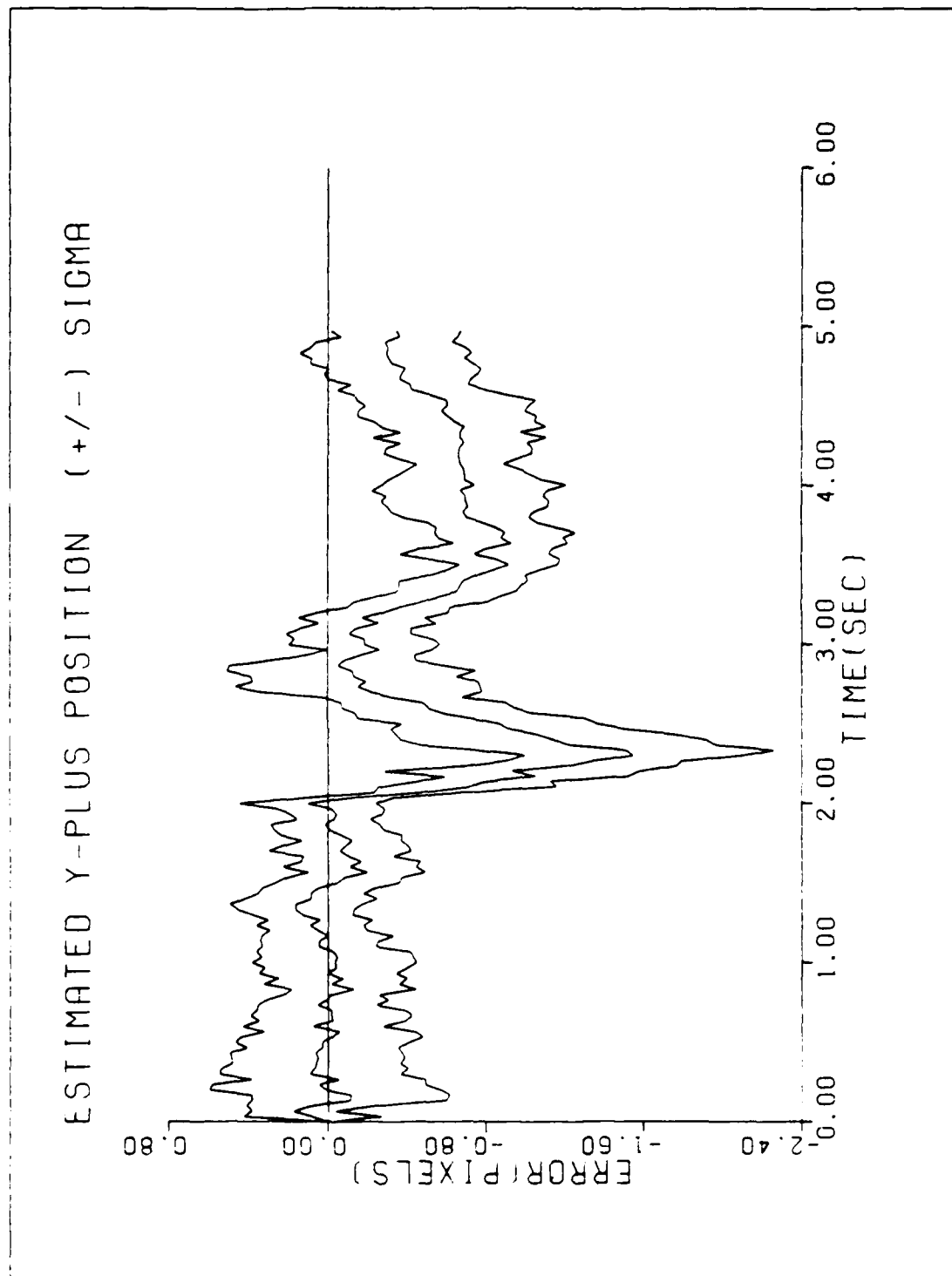


Figure F-30. $GM/MMAF/T2/20-G/\tau=4,.4$

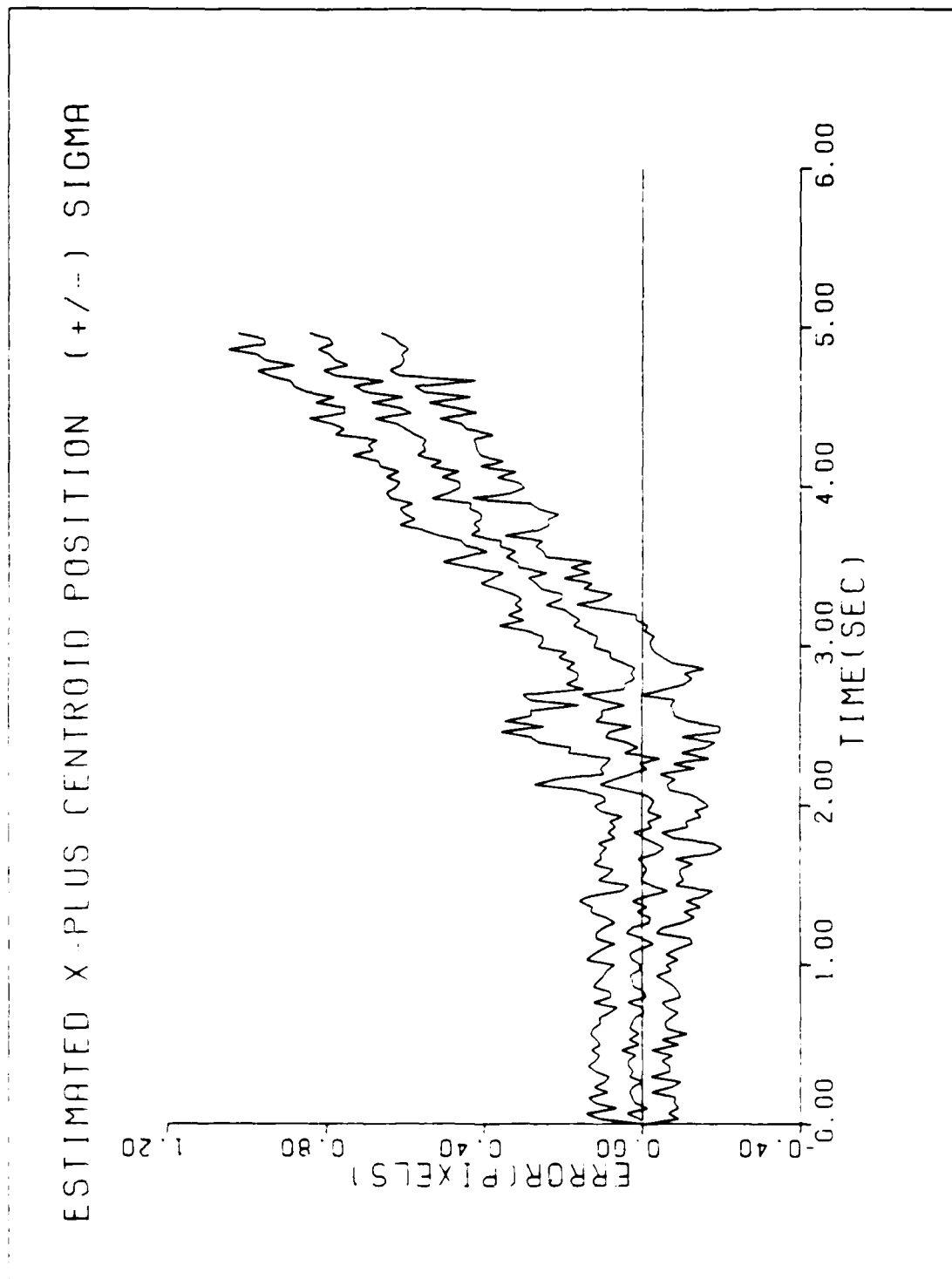


Figure F-31. $GM/MMAF/T2/20-G/\tau=4,.4$

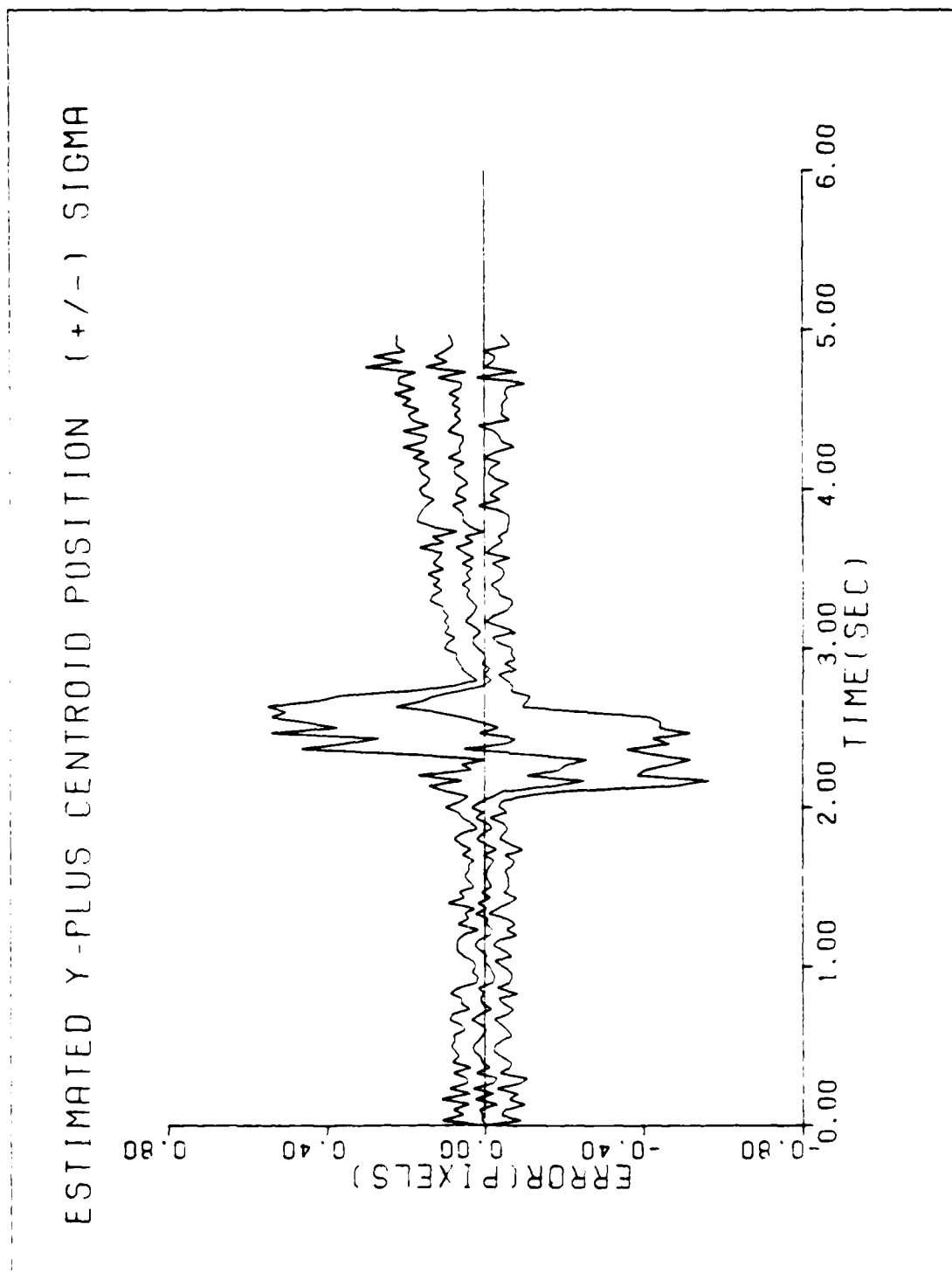


Figure F-32. GM/MMAF/T2/20-6/ $\tau=4.4$

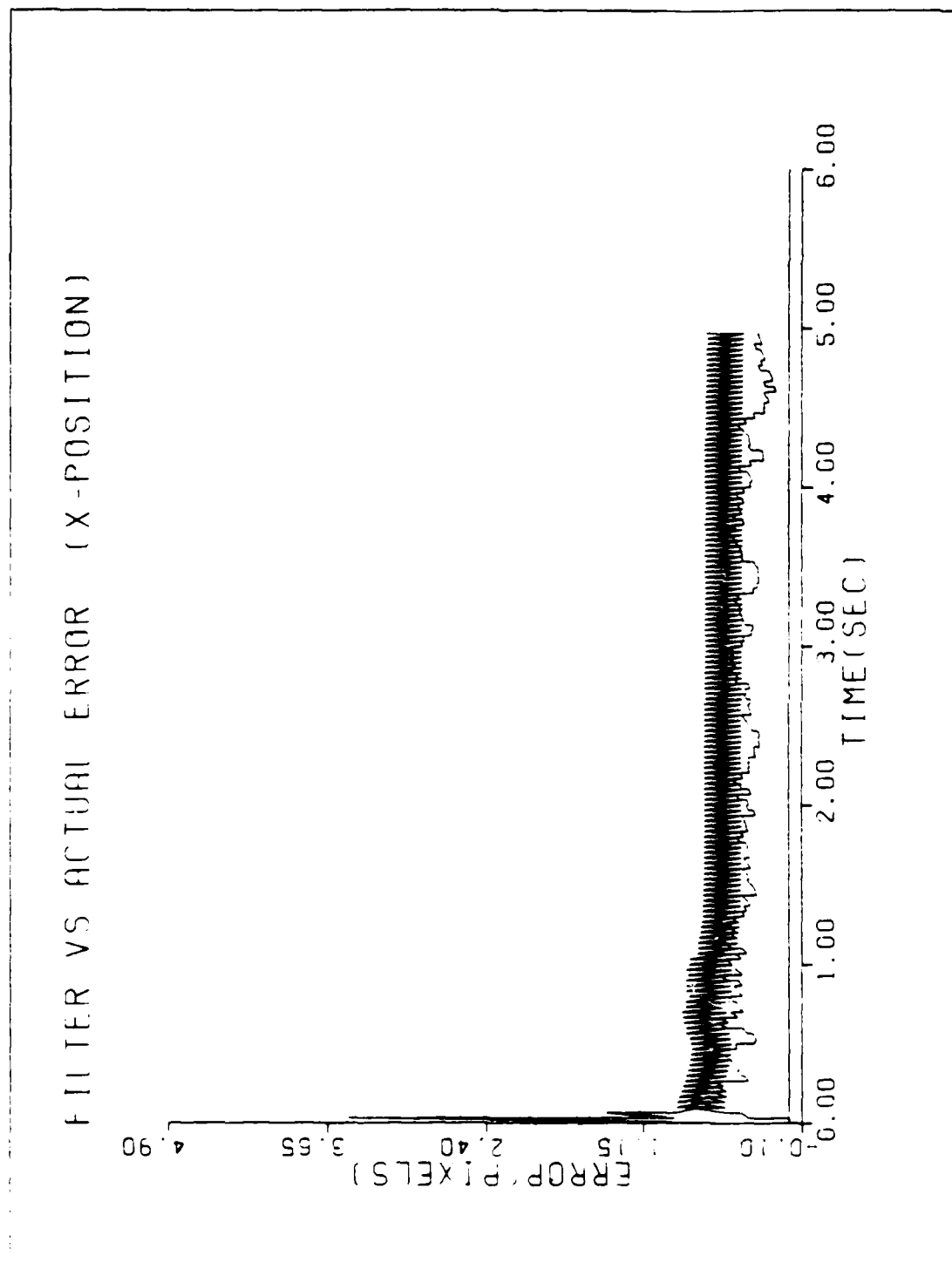


Figure F-33. $GM/MMAF/T2/10-G/\tau=4, .4/KP=2$ mrad

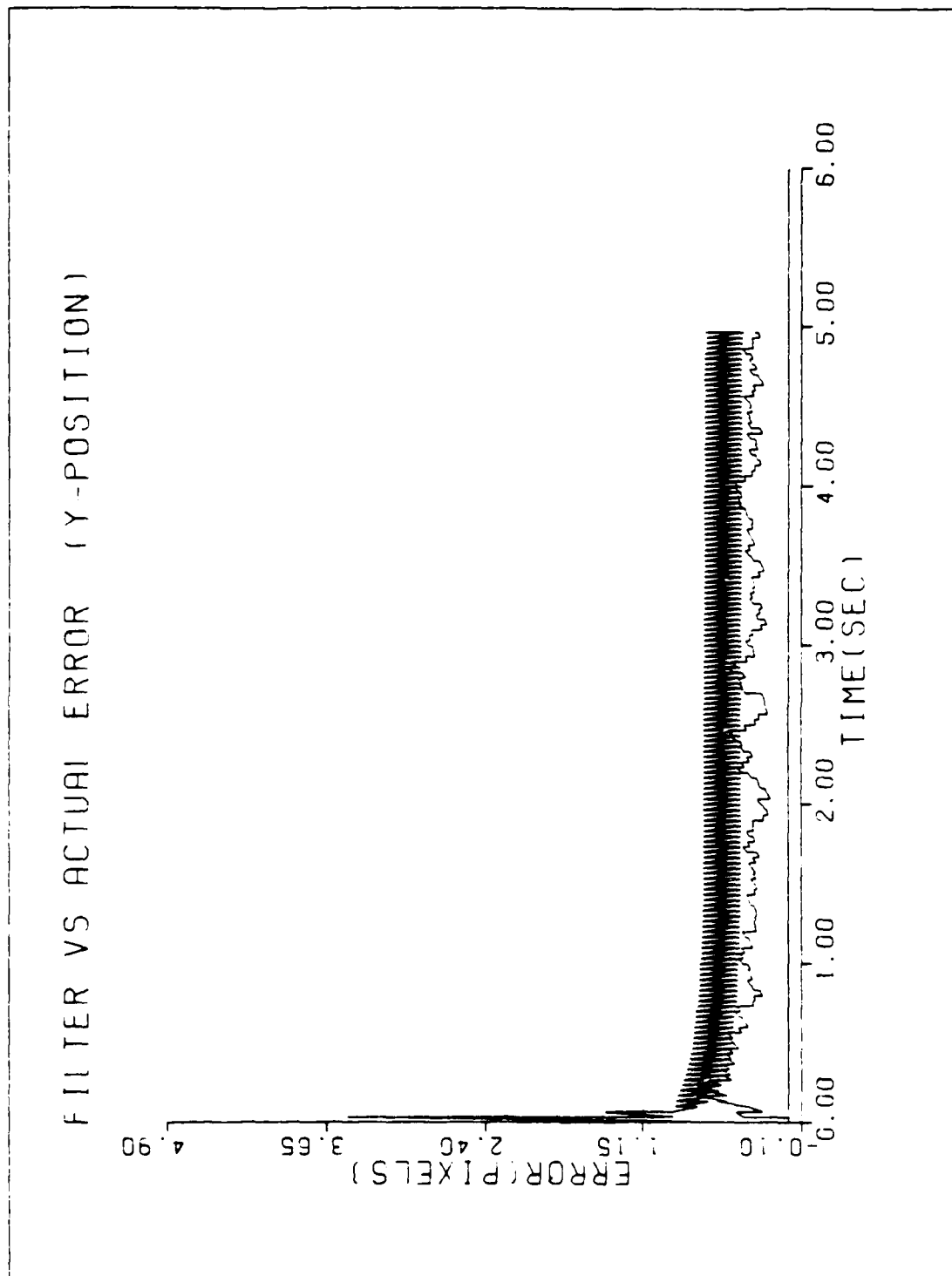


Figure F-34. GM/MMAF/T2/10-G/ $\tau=4.4$ /KP=2 mrad

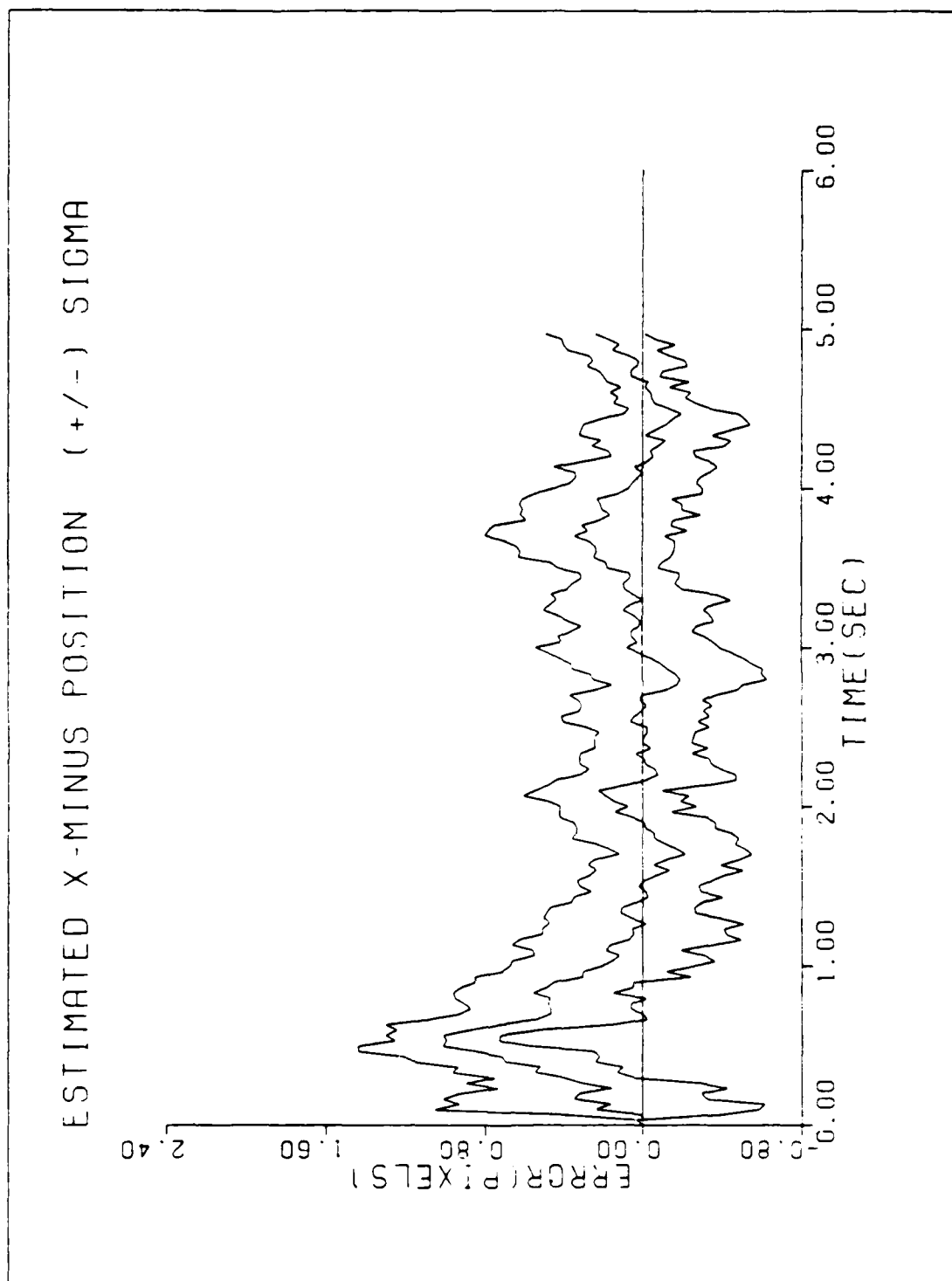


Figure F-35. $GM/MMAF/T2/10-G/\tau=4, .4/KP=2$ mrad

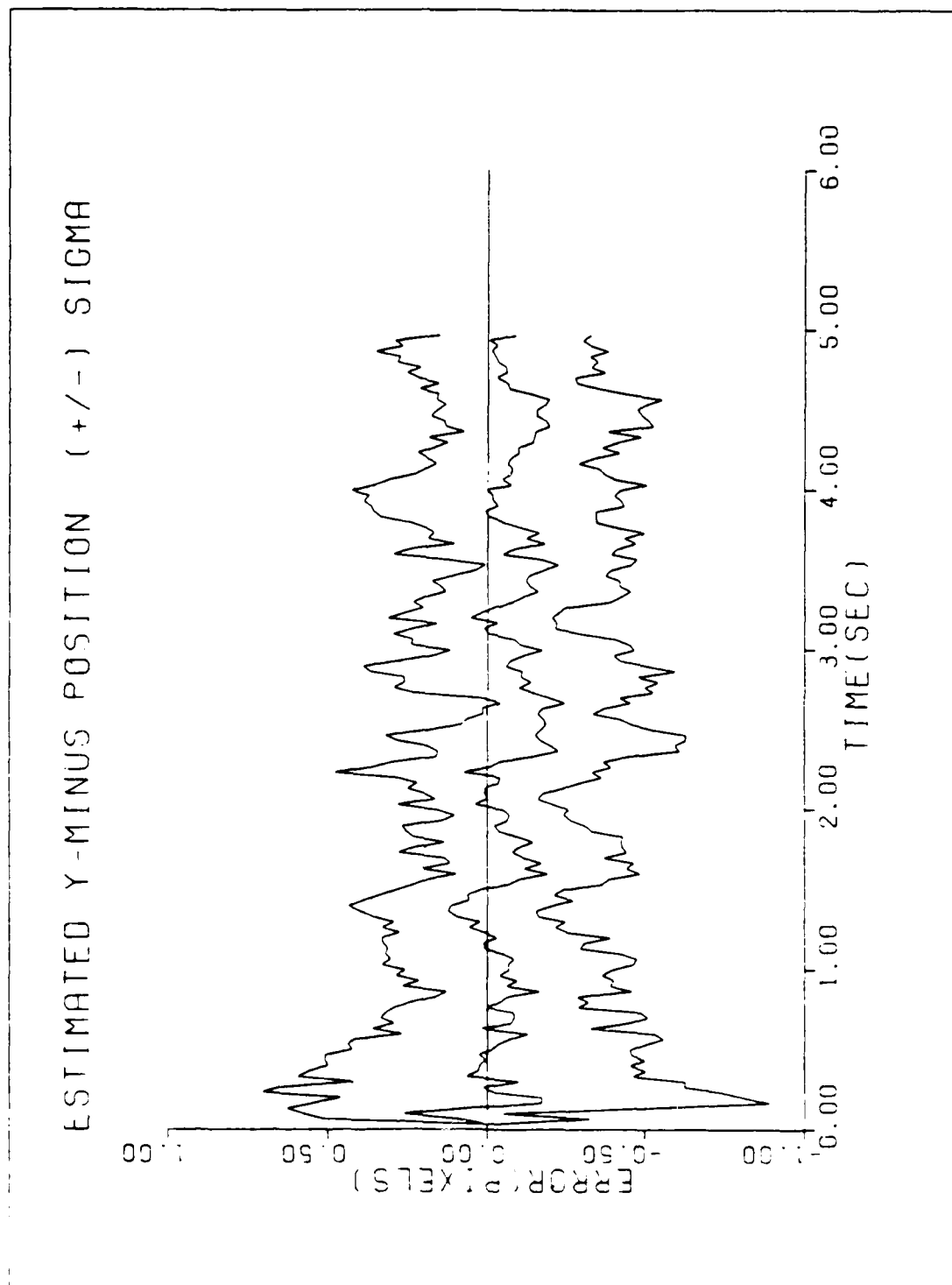


Figure F-36. $GM/MMAF/T2/10-G/\tau=4,.4/KP=2$ mrad

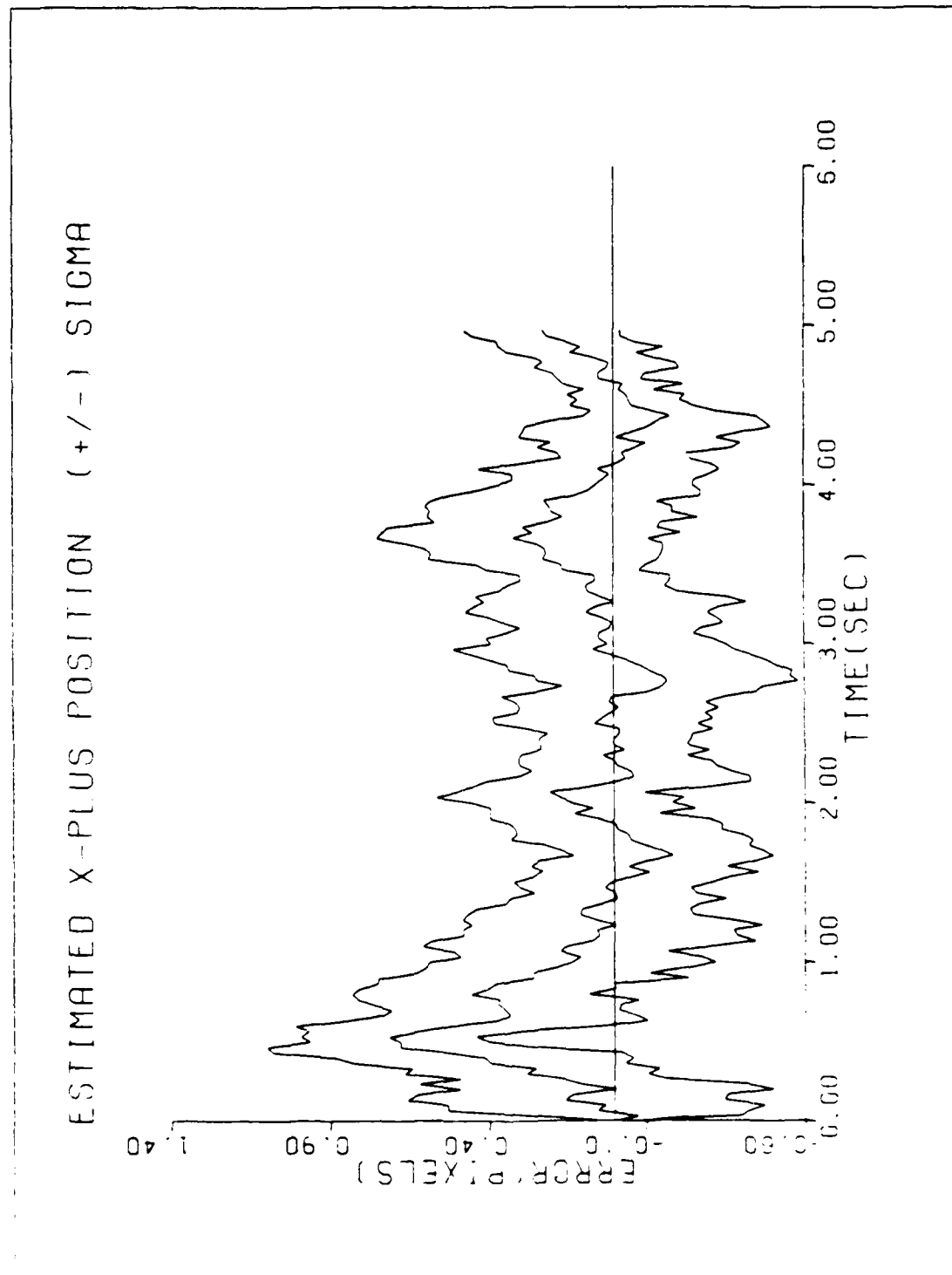


Figure F-37. $GM/MMAF/T2/10-G/\tau=4, .4/KP=2$ mrad

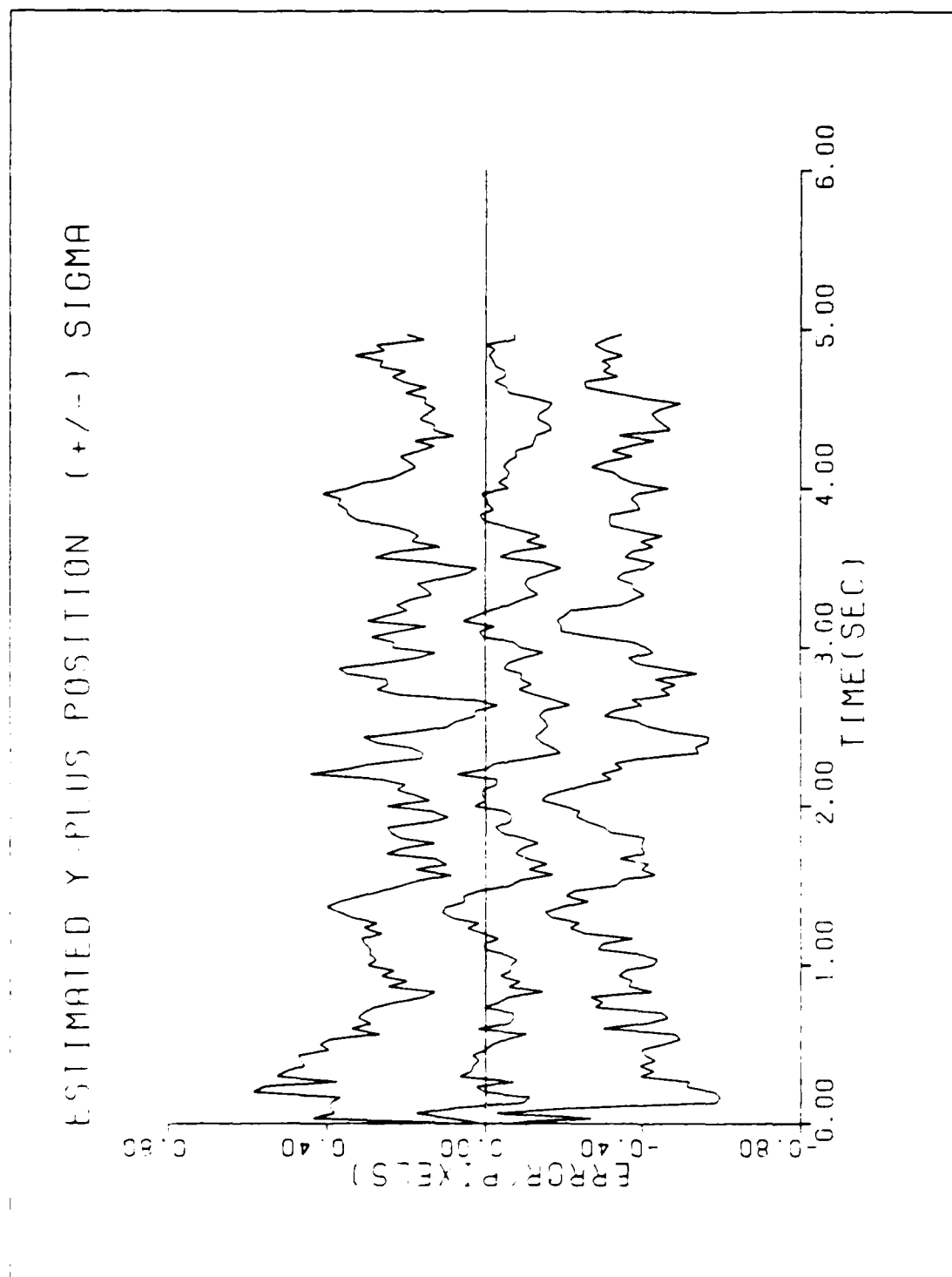


Figure F-38. $GM/MMAF/T2/10-G/\tau=4, .4/KP=2$ mrad

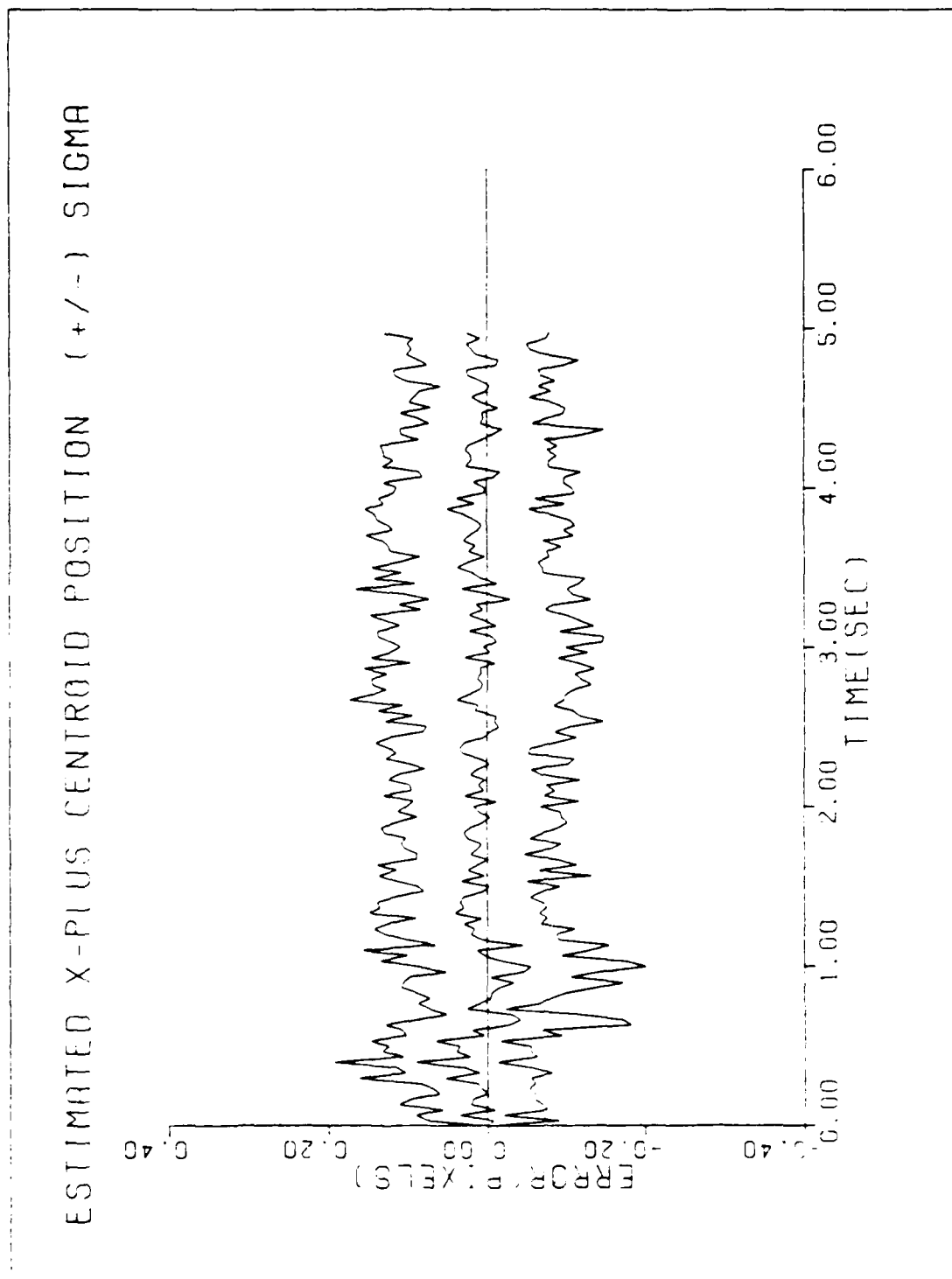


Figure F-39. $GM/MMAF/T2/10-G/\tau=4,.4/KP=2$ mrad

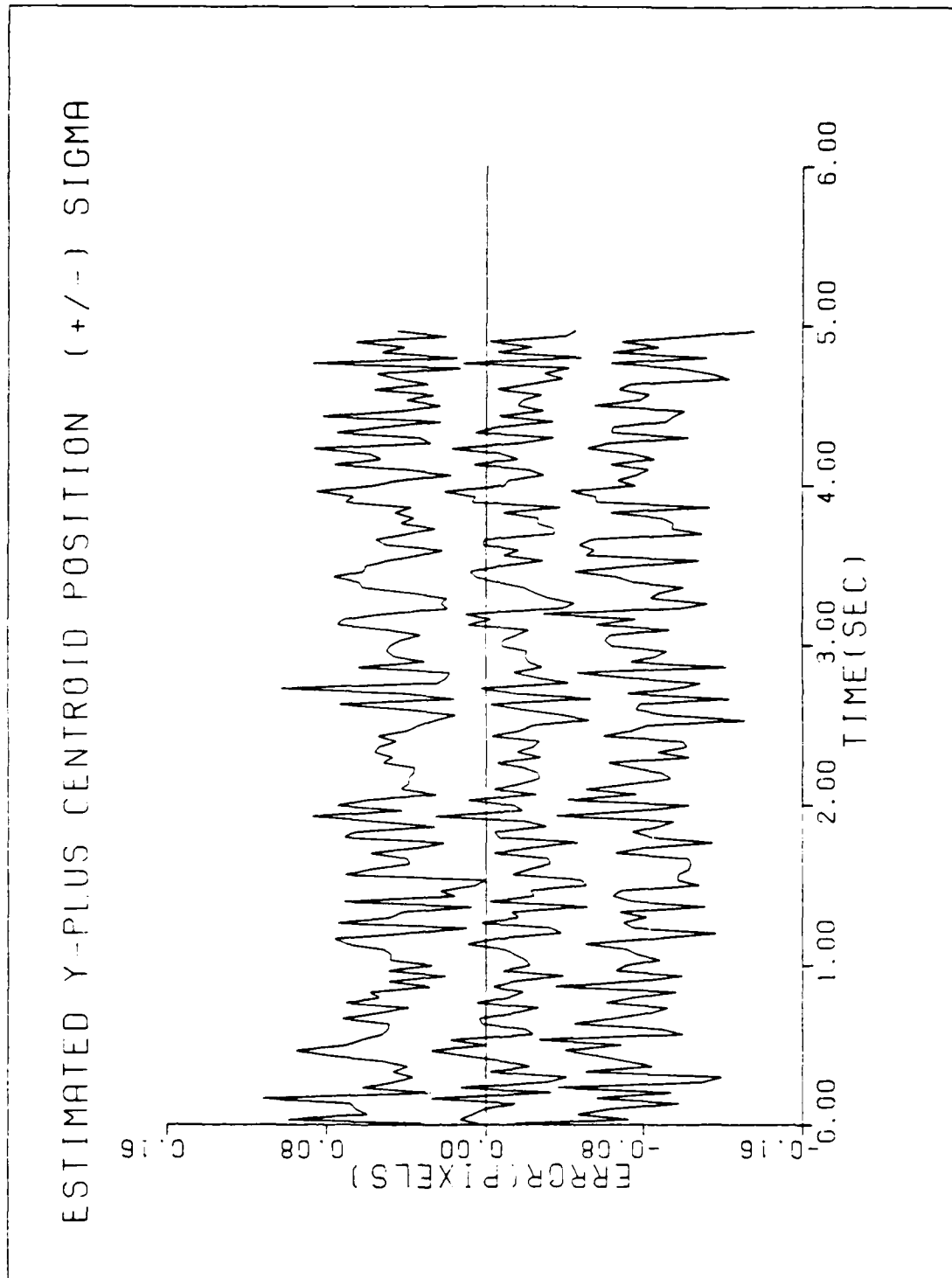


Figure F-40. $GM/MMAF/T2/10-G/\tau=4,.4/KP=2$ mrad

APPENDIX G

**MULTIPLE MODEL ADAPTIVE FILTER SIMULATIONS
TARGET TRAJECTORY SENSITIVITY ANALYSIS
(REFERENCED FROM SECTION 6.6)**

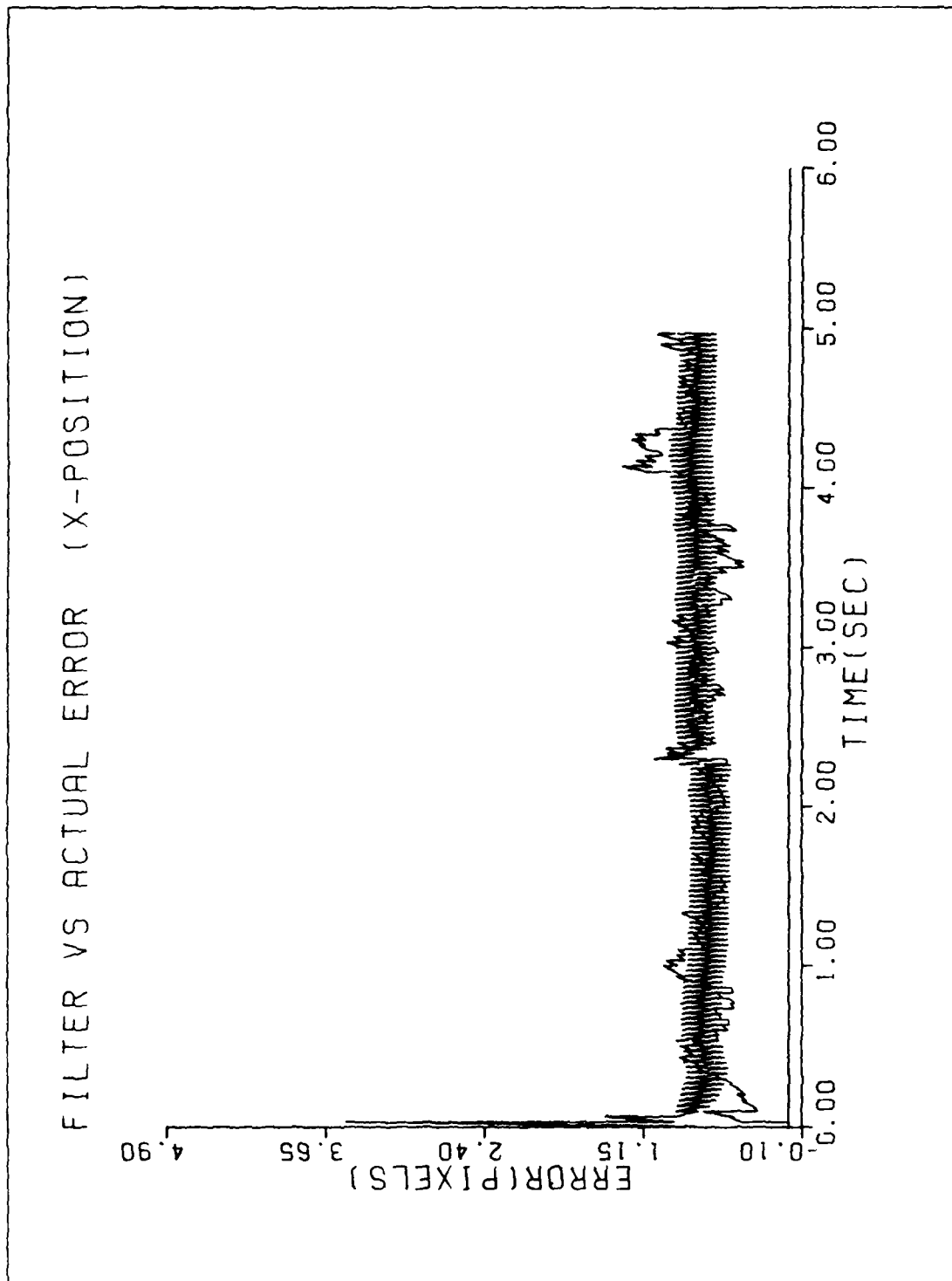


Figure G-1. GM/MMAF/T5/10-G/QB1

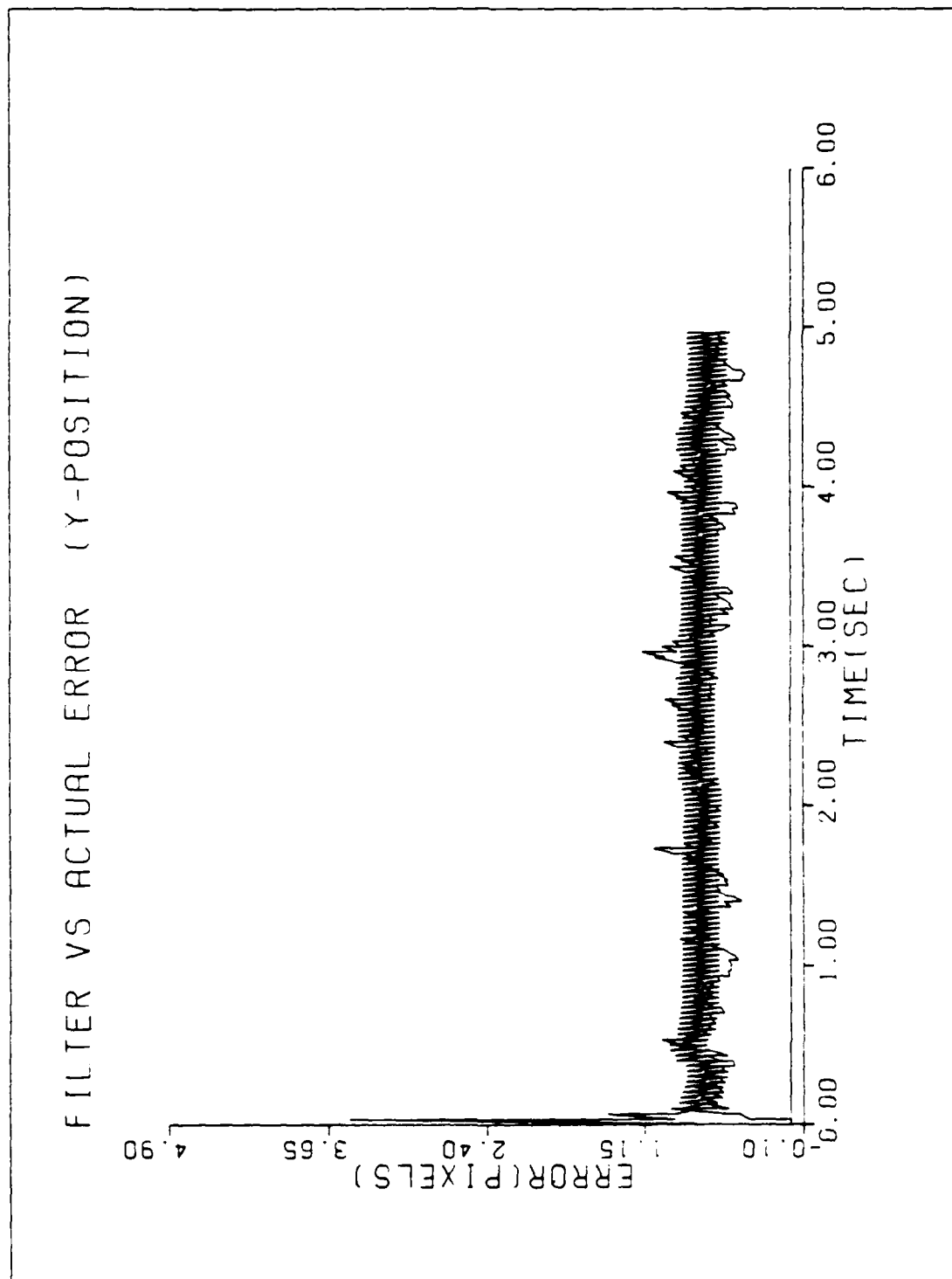


Figure 6-2. GM/MMAF/T5/10-G/QB1

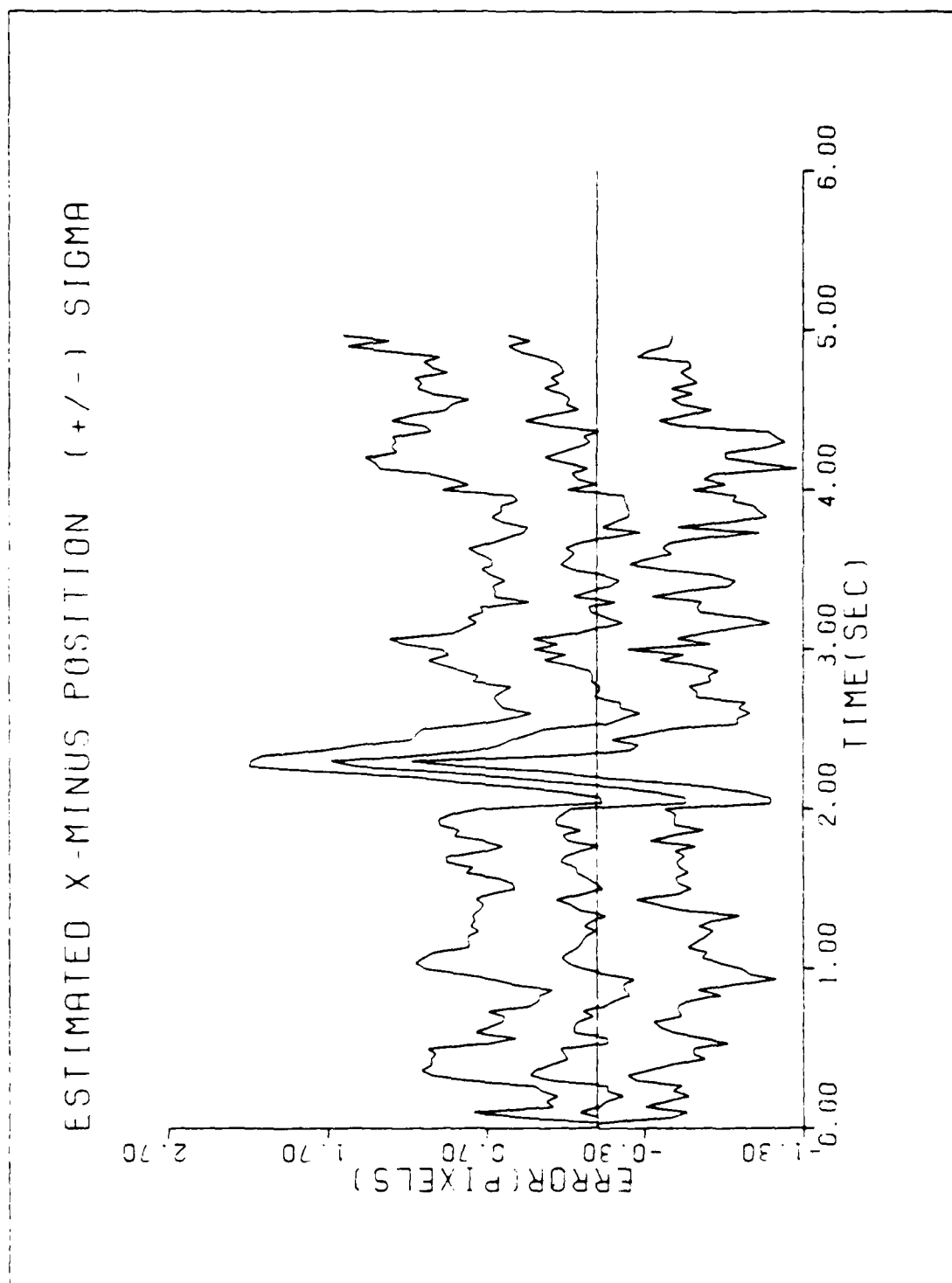


Figure 6-3. GM/MMAF/T5/10-6/QB1

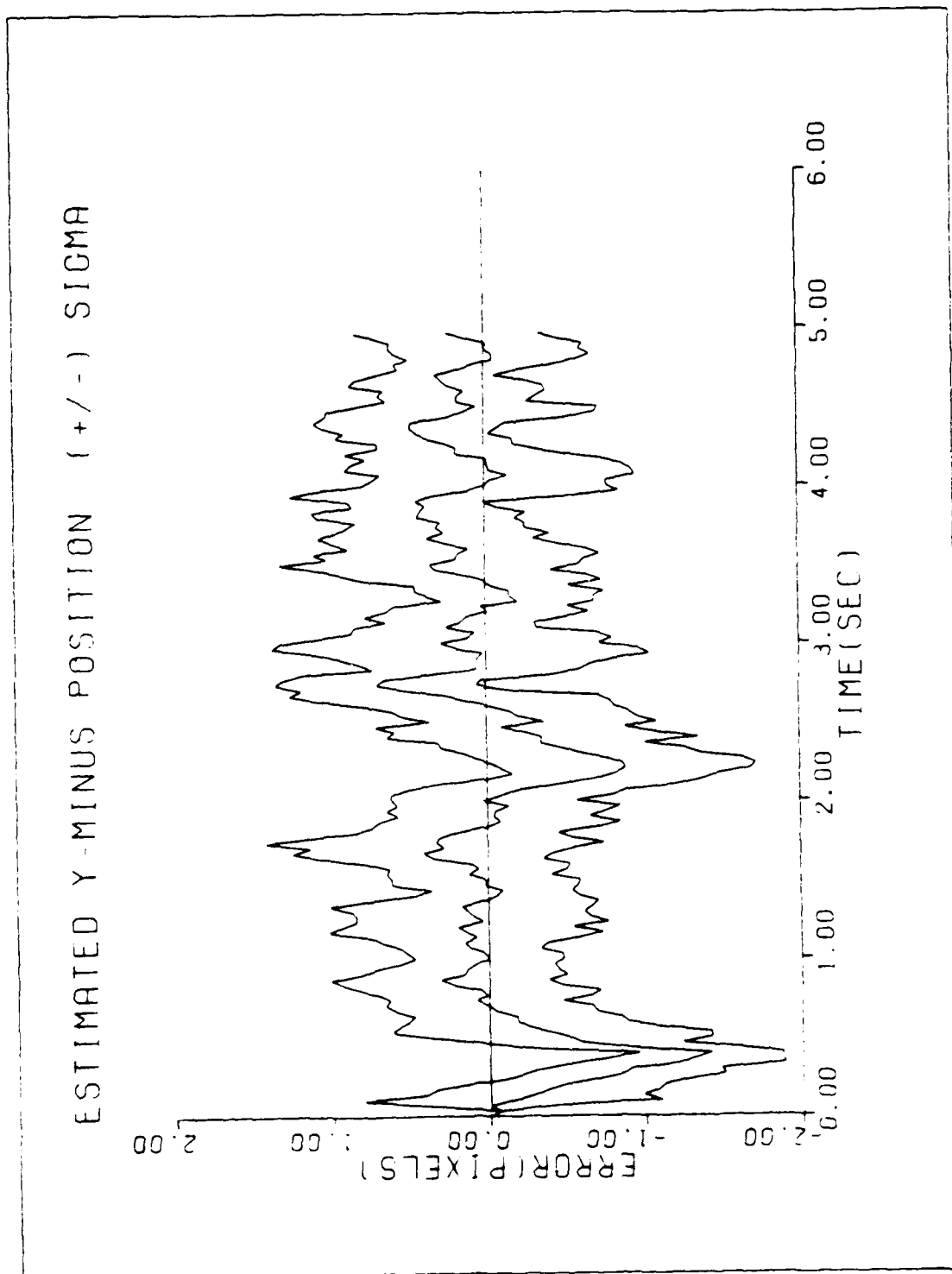


Figure G-4. GM/MMAF/T5/10-G/QB1

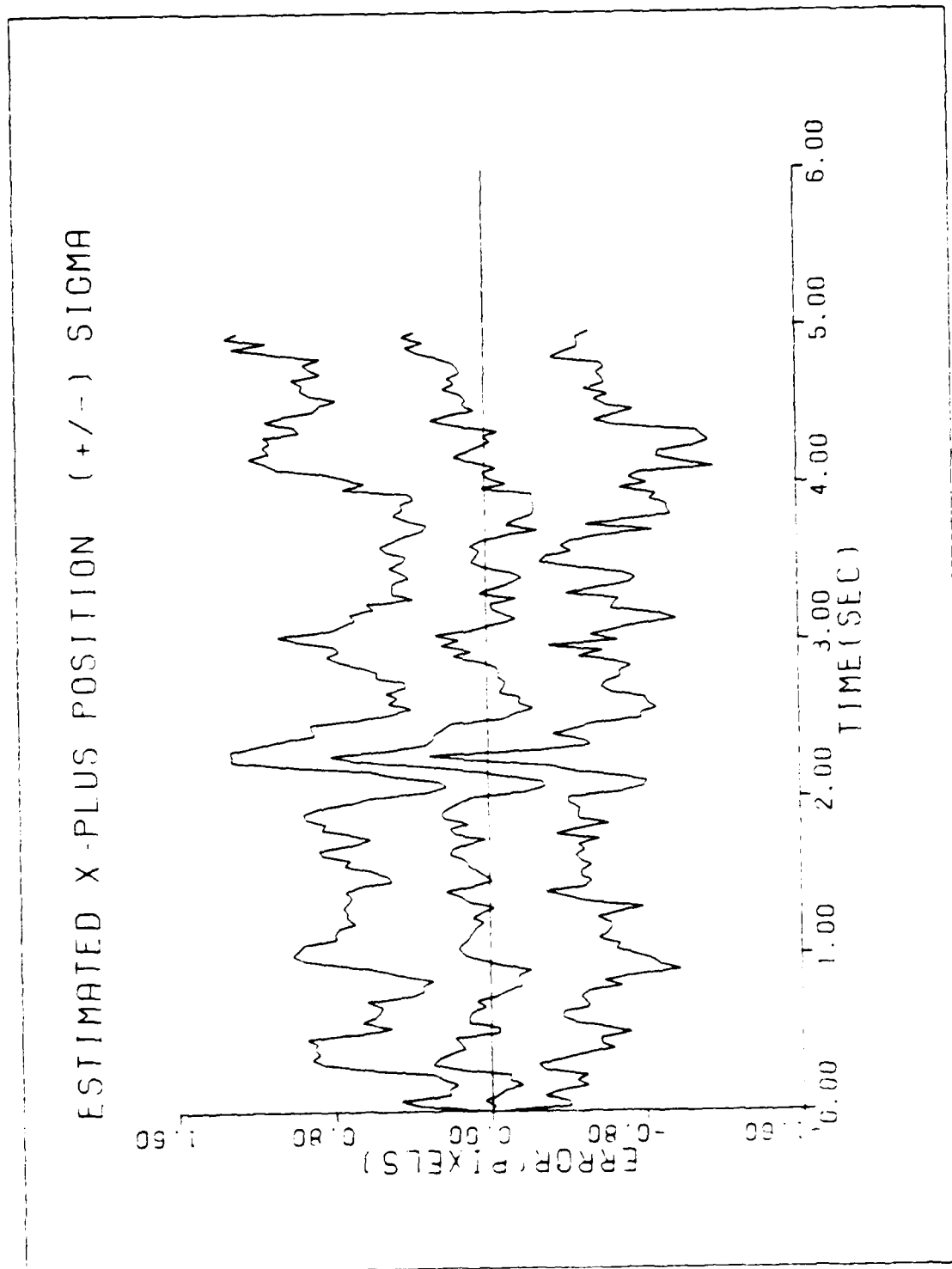


Figure G-5. GM/MMAF/T5/10-G/QB1

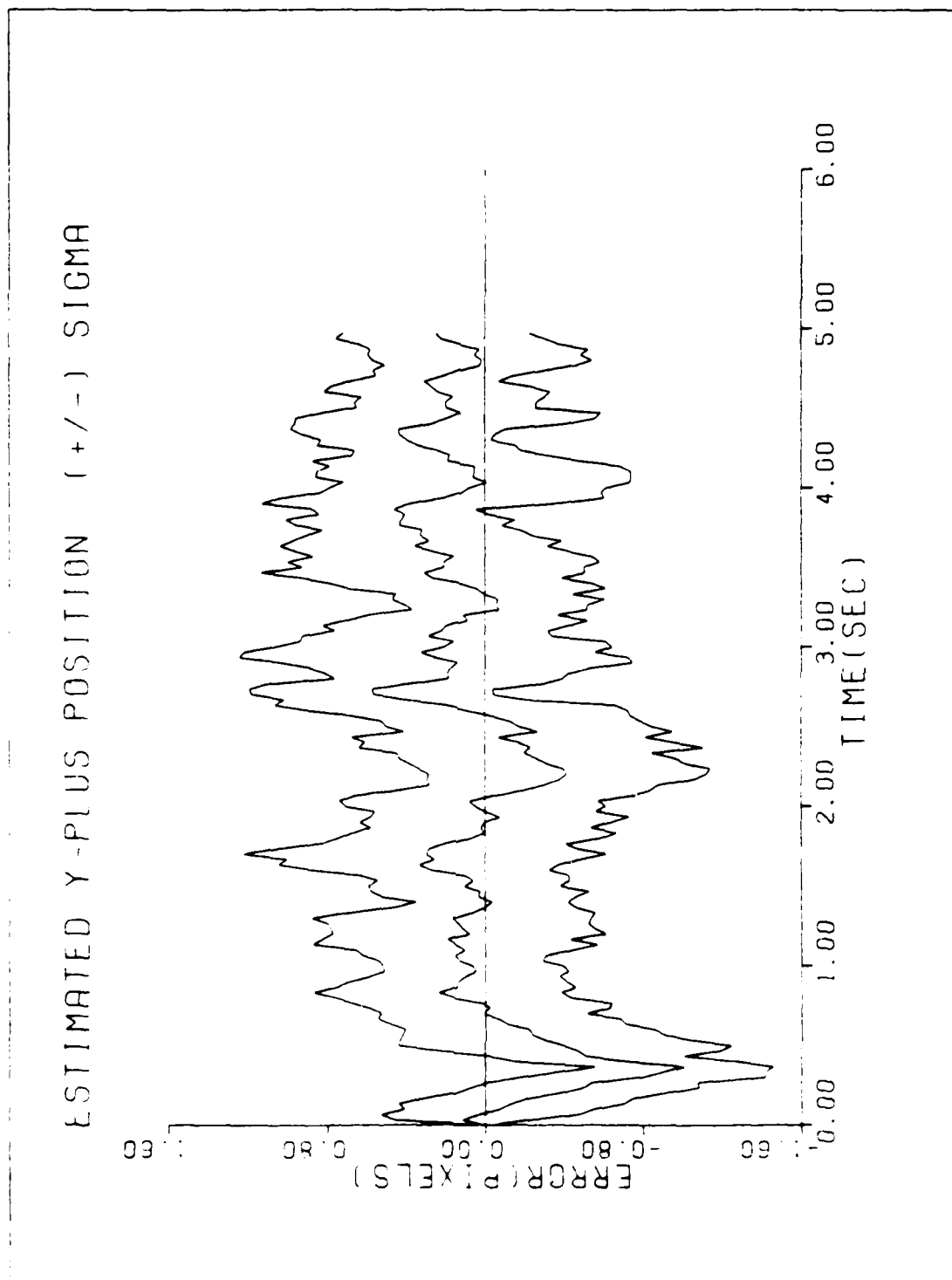


Figure G-6. GM/MMAF/T5/10-G/QB1

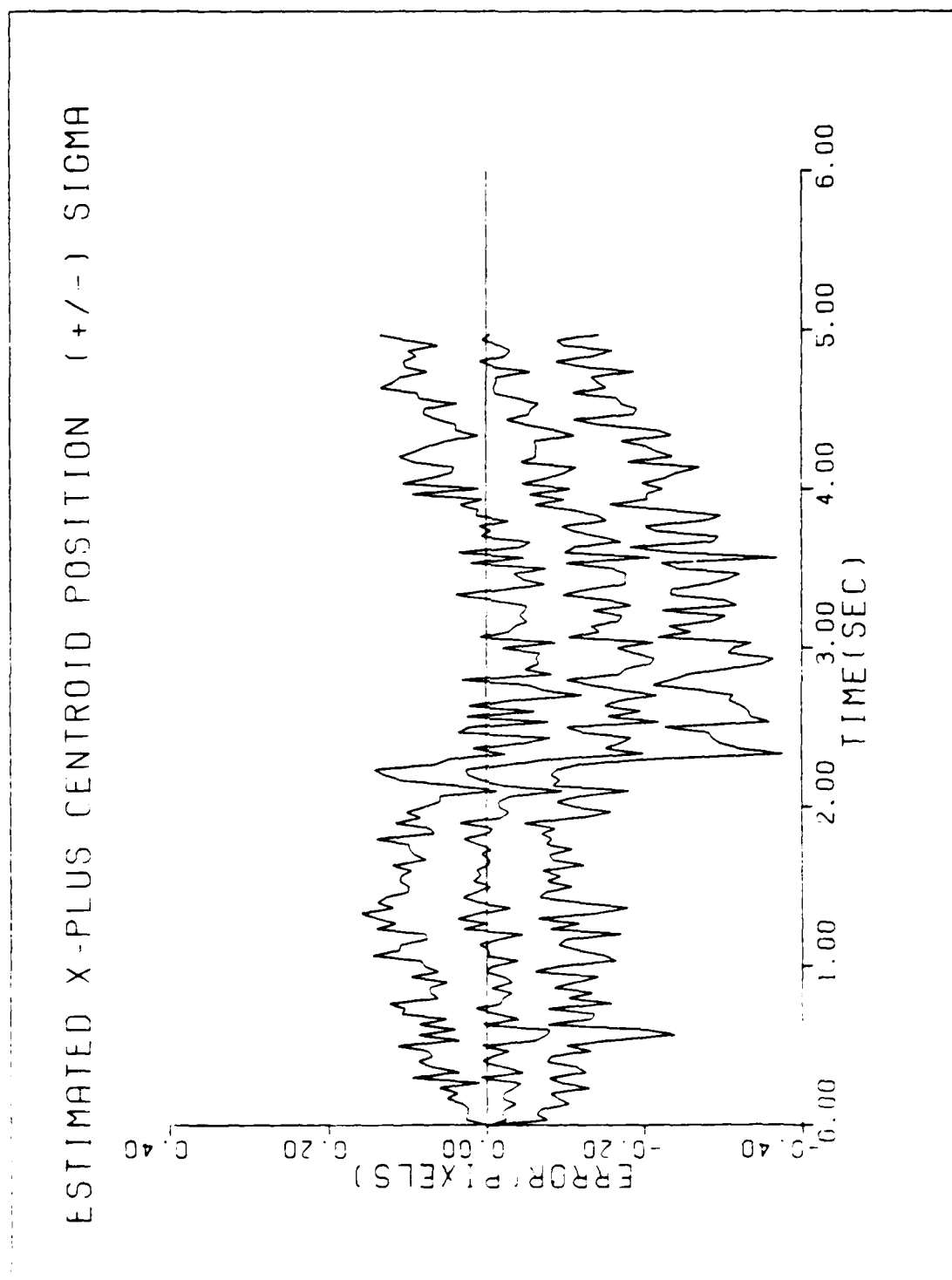


Figure G-7. GM/MMAF/T5/10-G/QB1

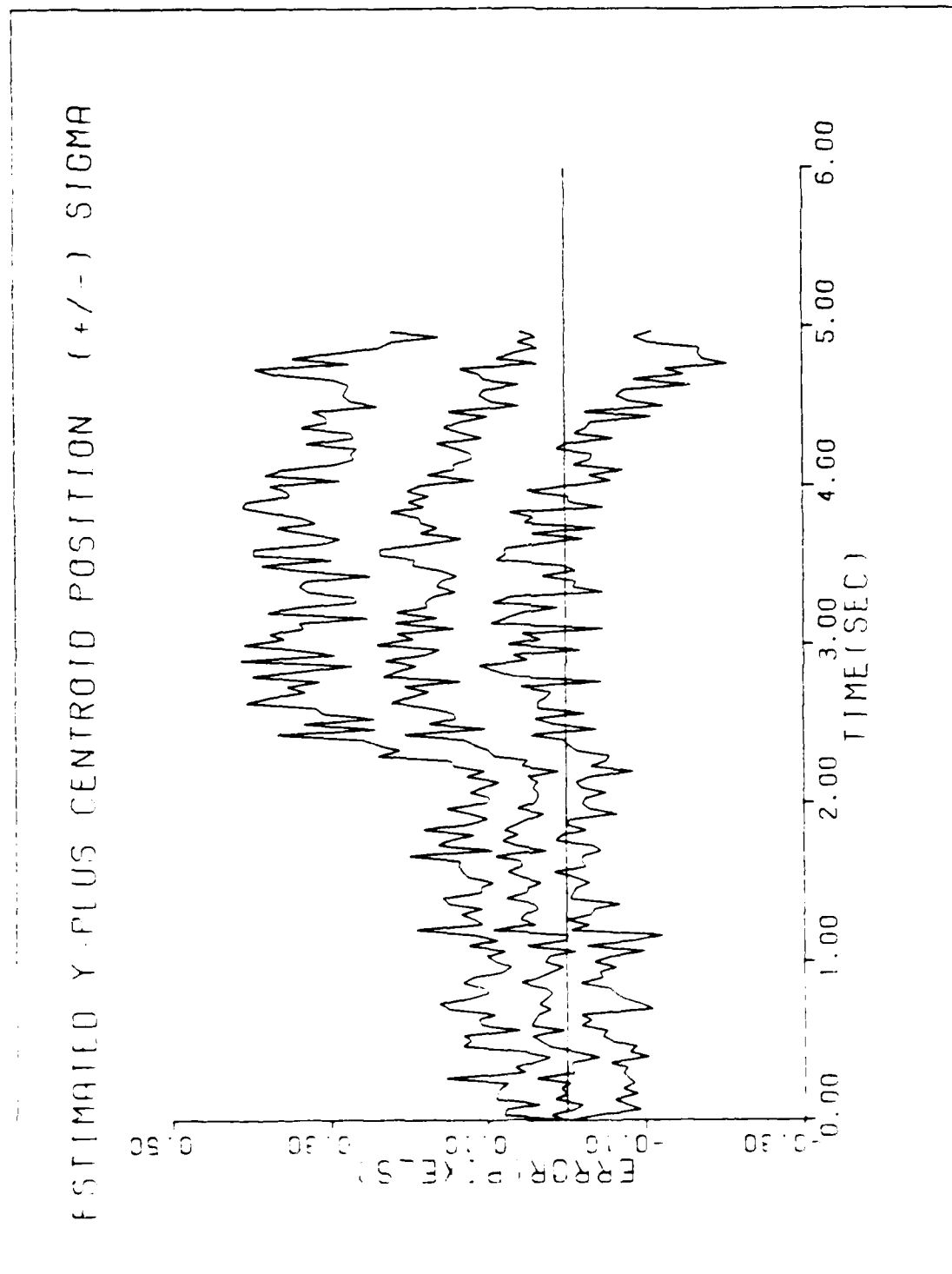


Figure G-8. GM/MMAF/T5/10-G/QB1

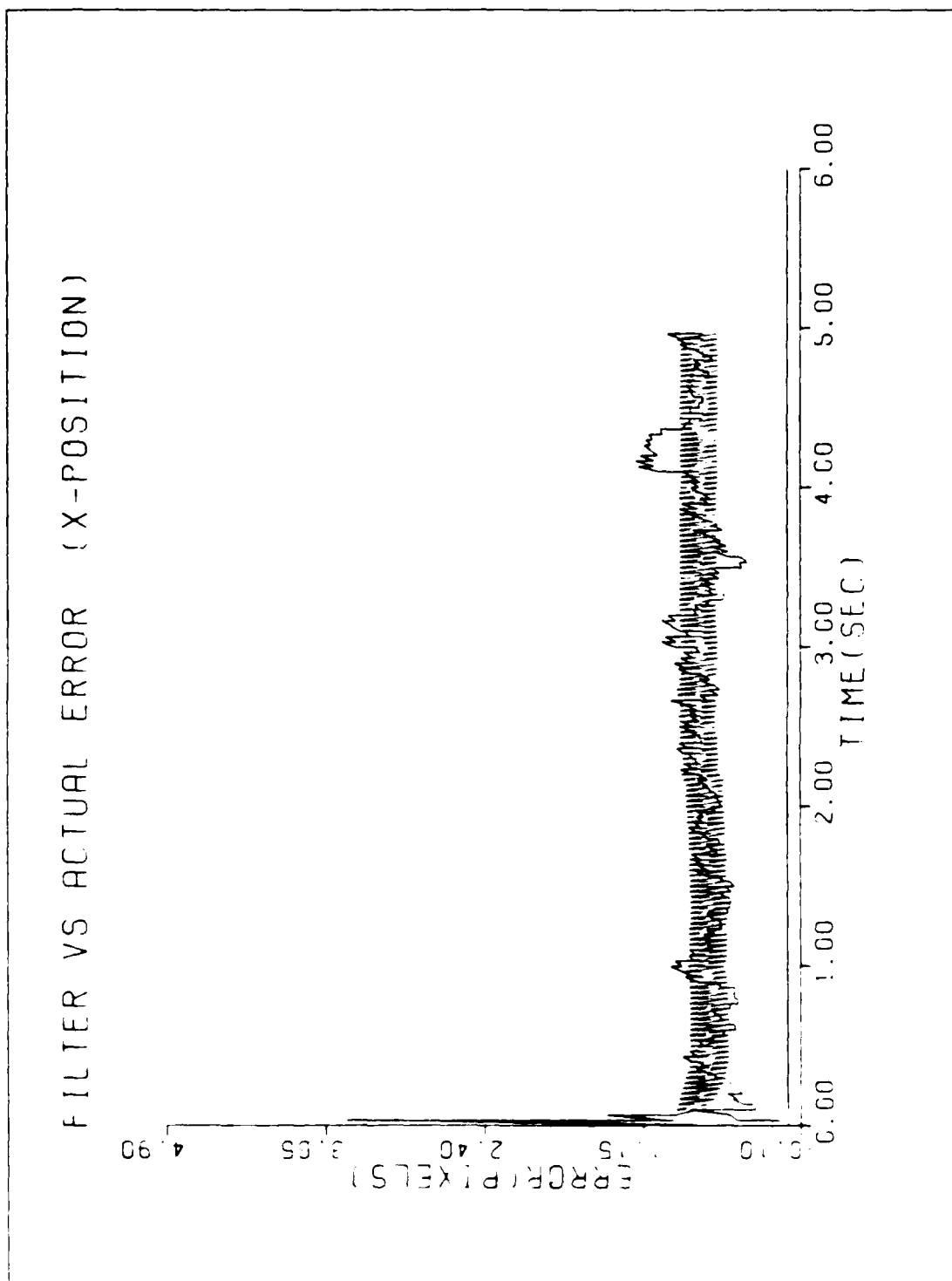


Figure 6-9. GM/MMAF/T6/10-G/QB1

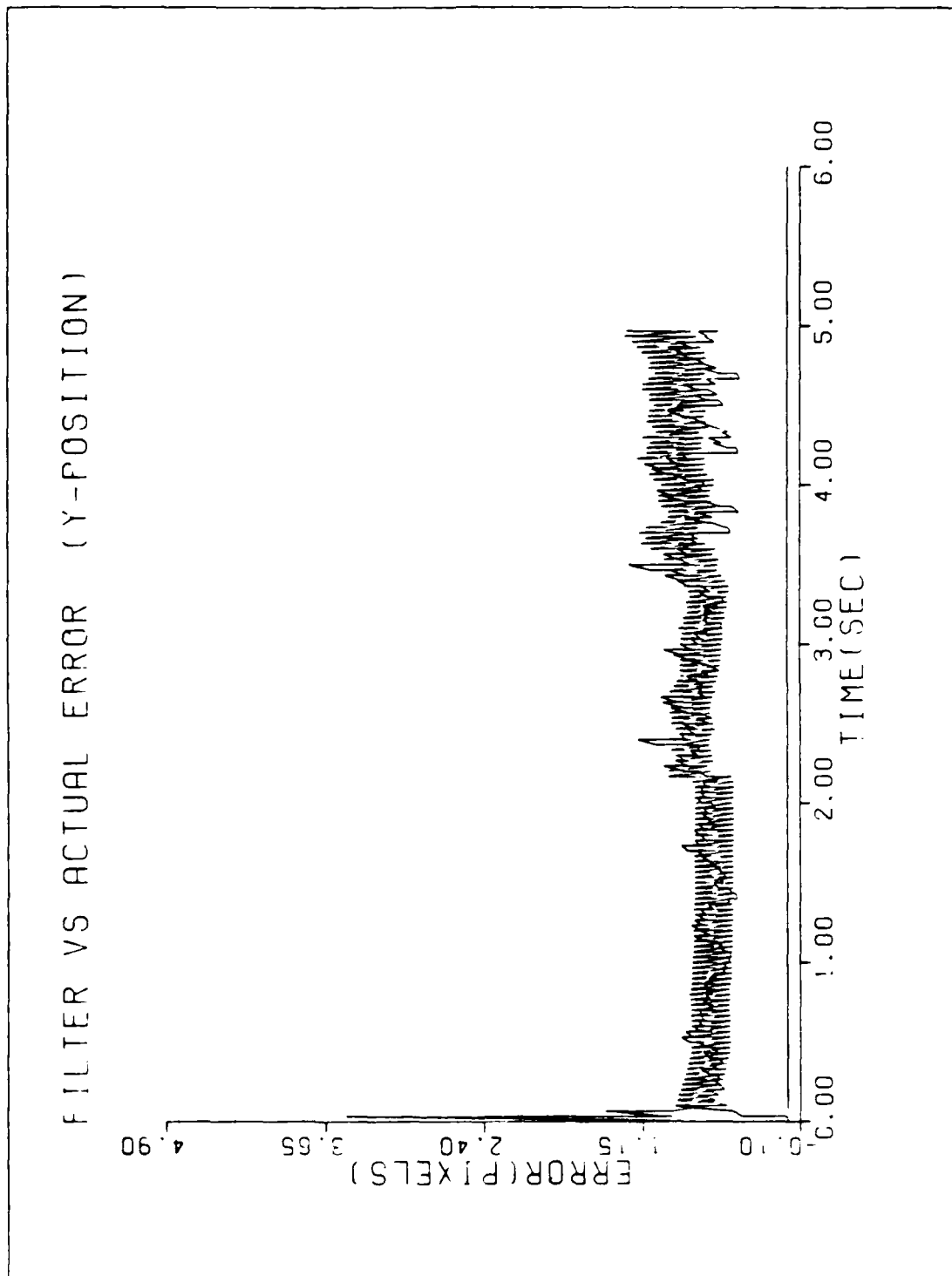


Figure 6-10. GM/MMAF/T6/10-G/QB1

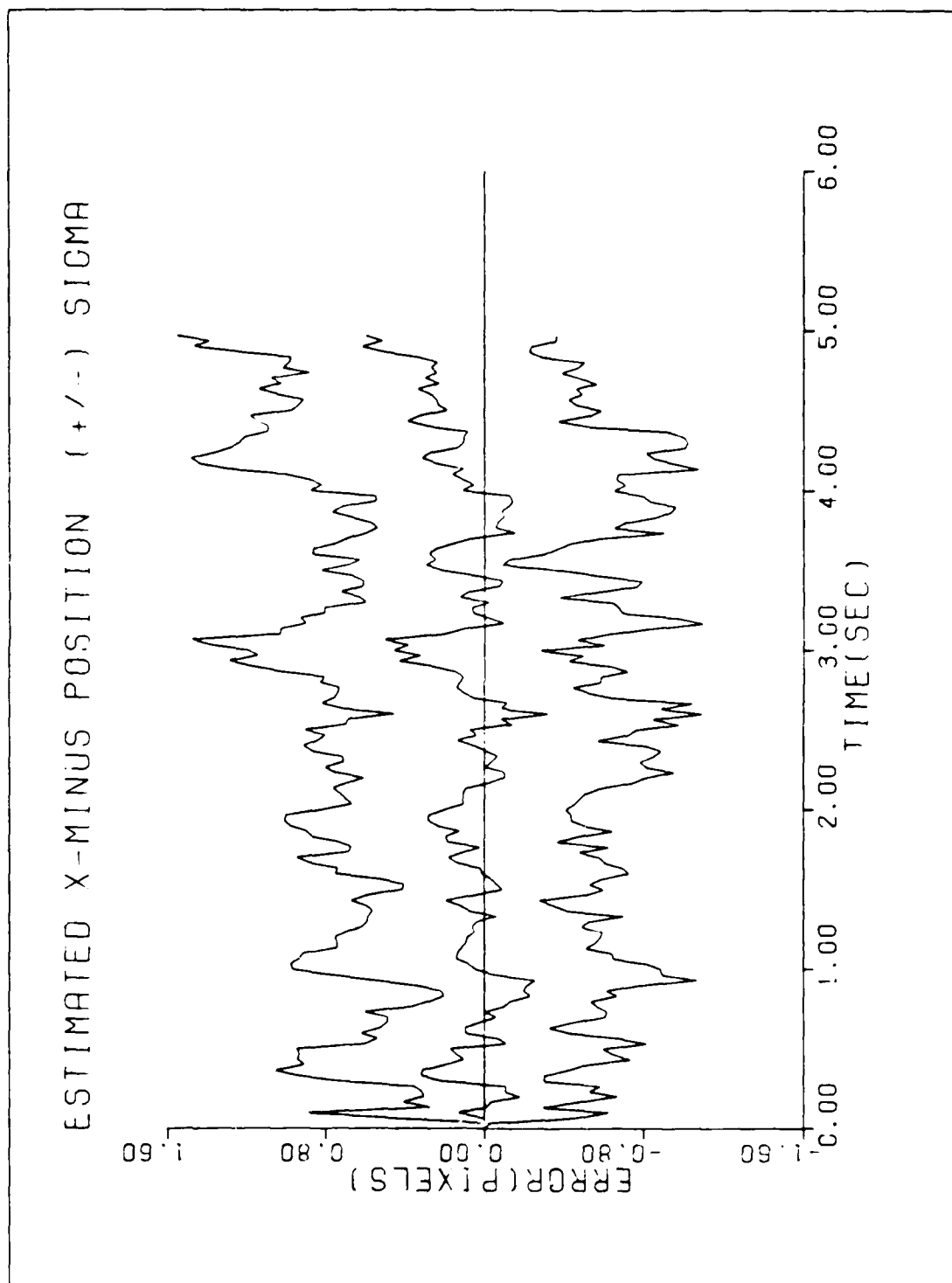


Figure G-11. GM/MMAF/T6/10-6/QB1

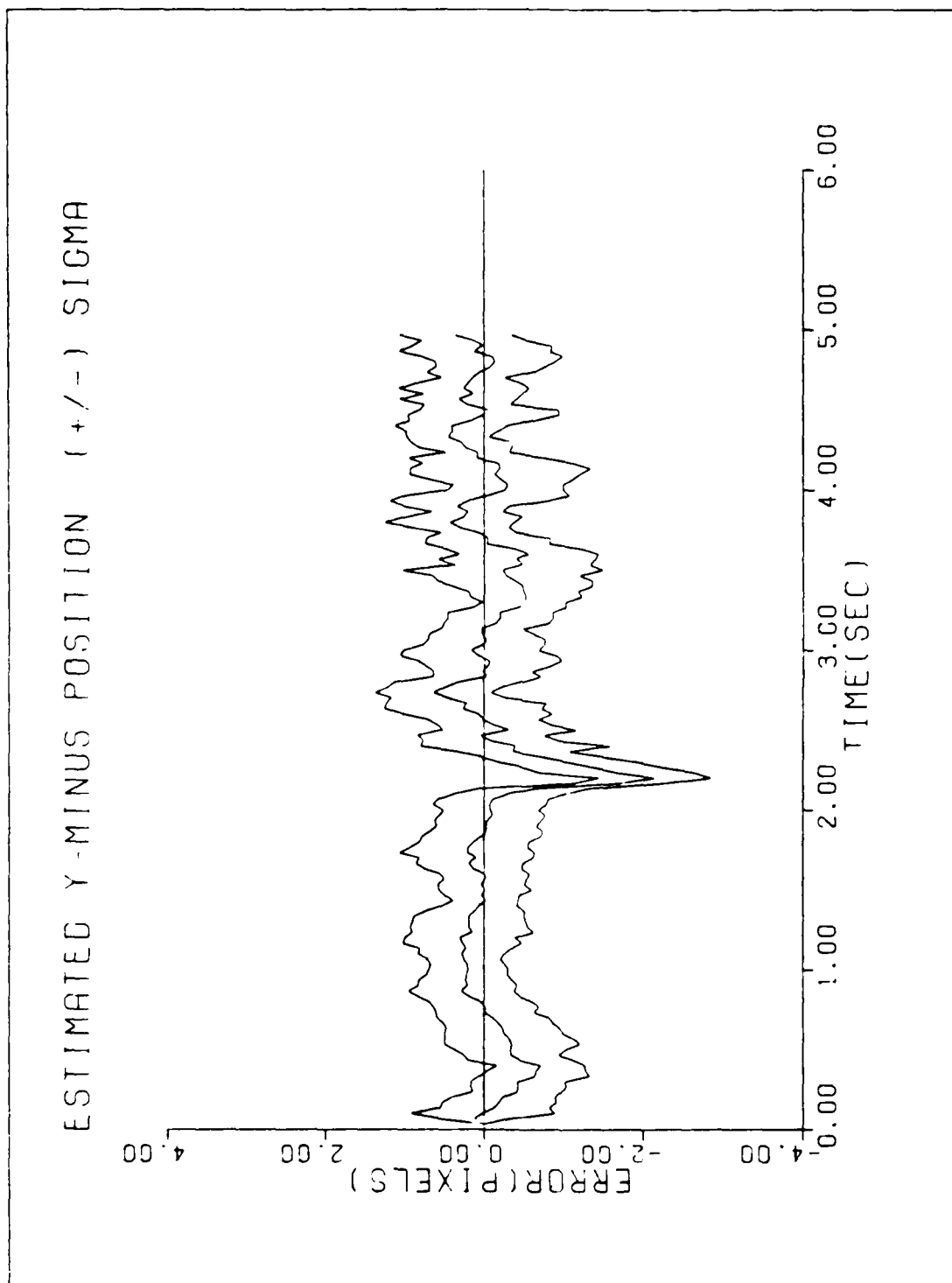


Figure G-12. GM/MMAF/T6/10-G/QB1

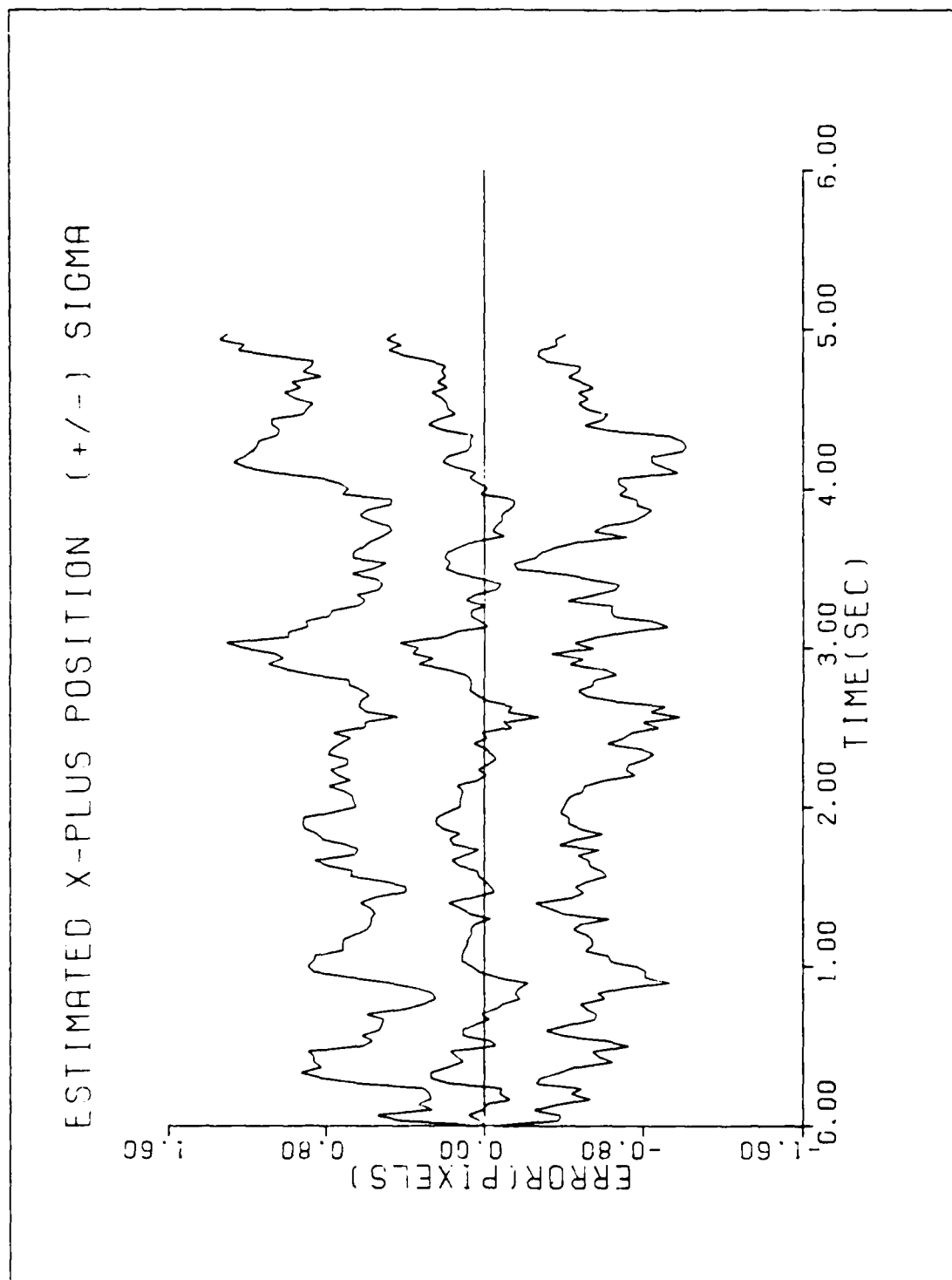


Figure G-13. GM/MMAF/T6/10-G/QB1

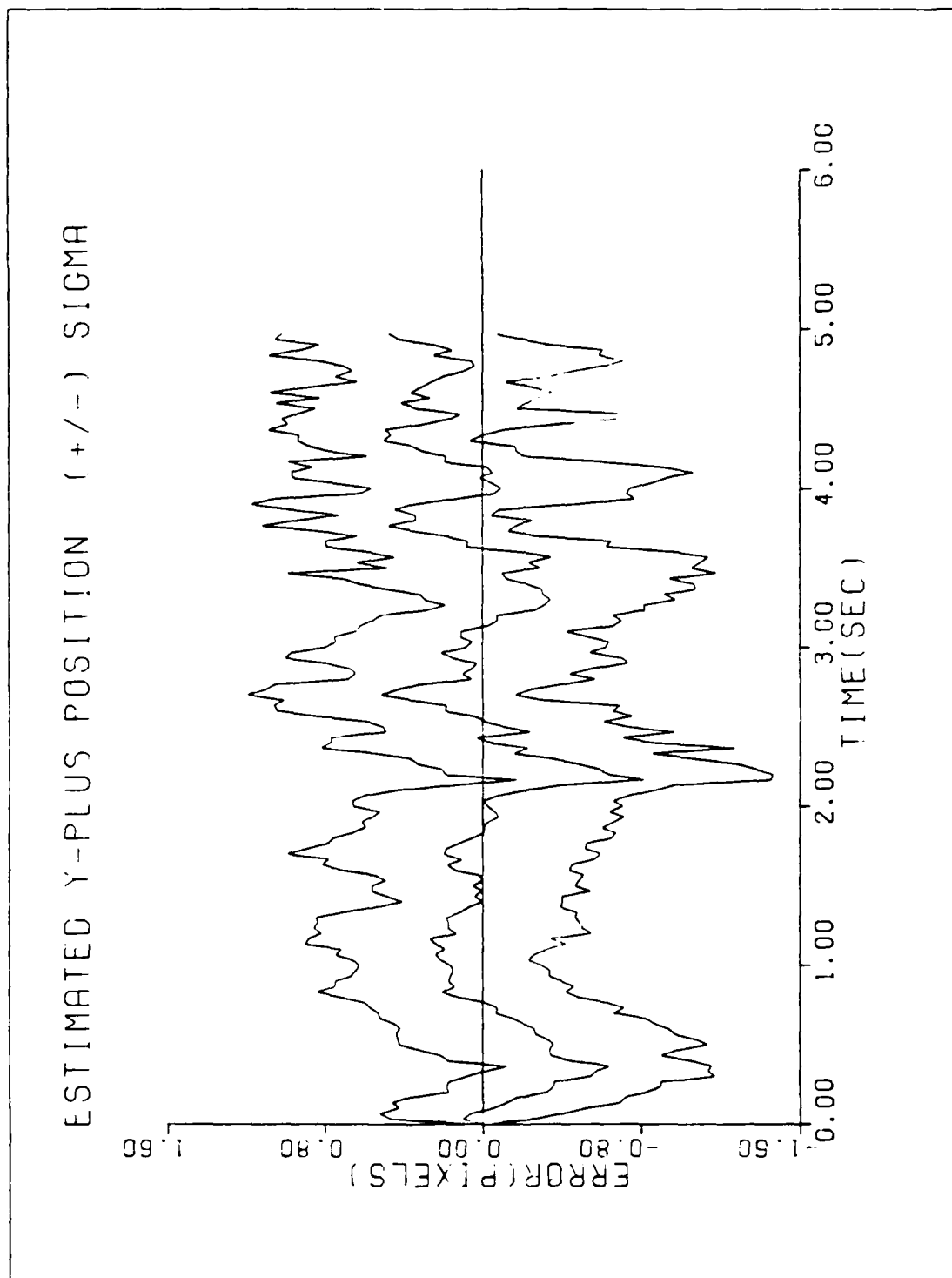


Figure G-14. GM/MMAF/T6/10-G/QB1

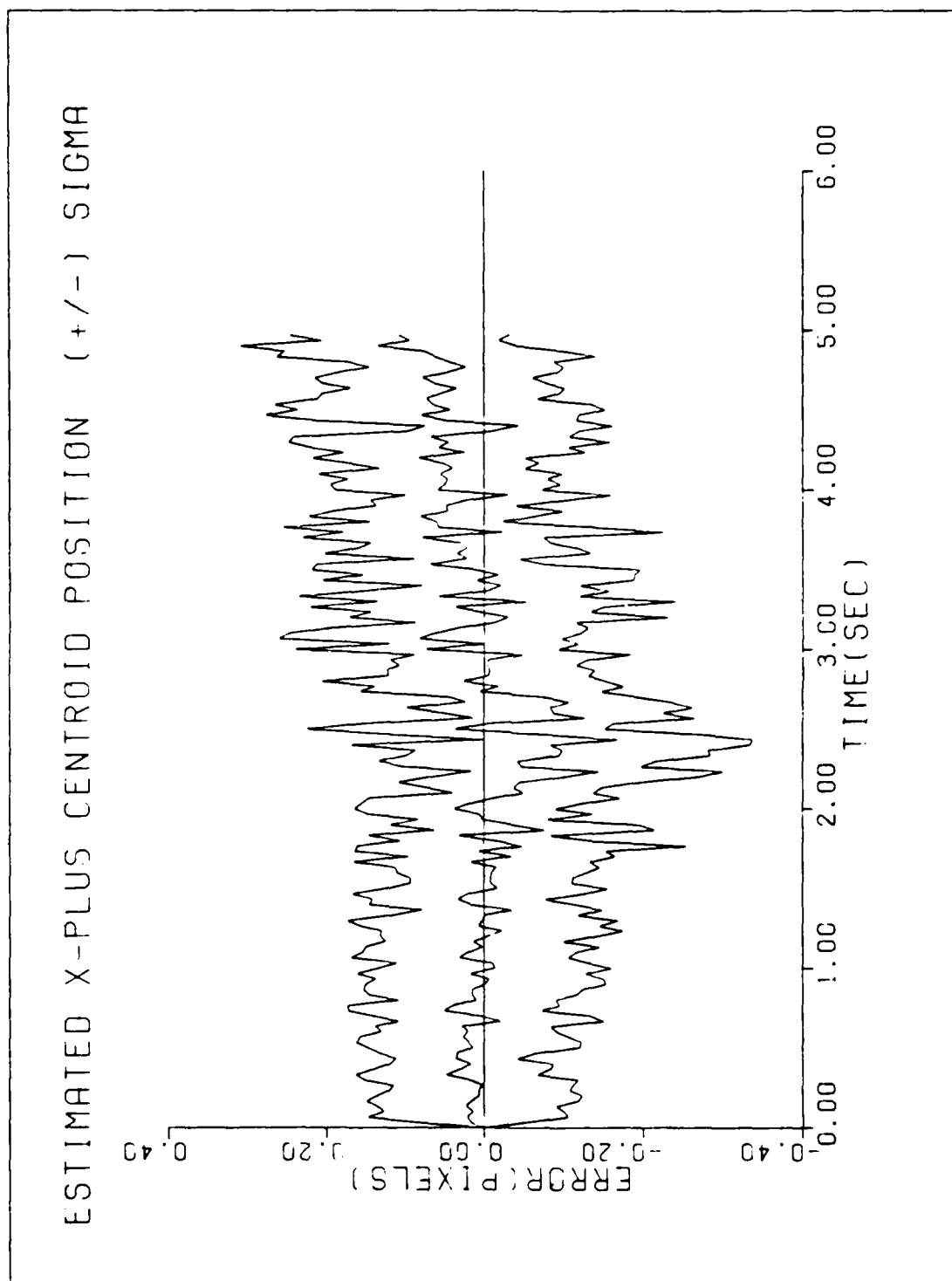


Figure 6-15. GM/MMAF/T6/10-G/QB1

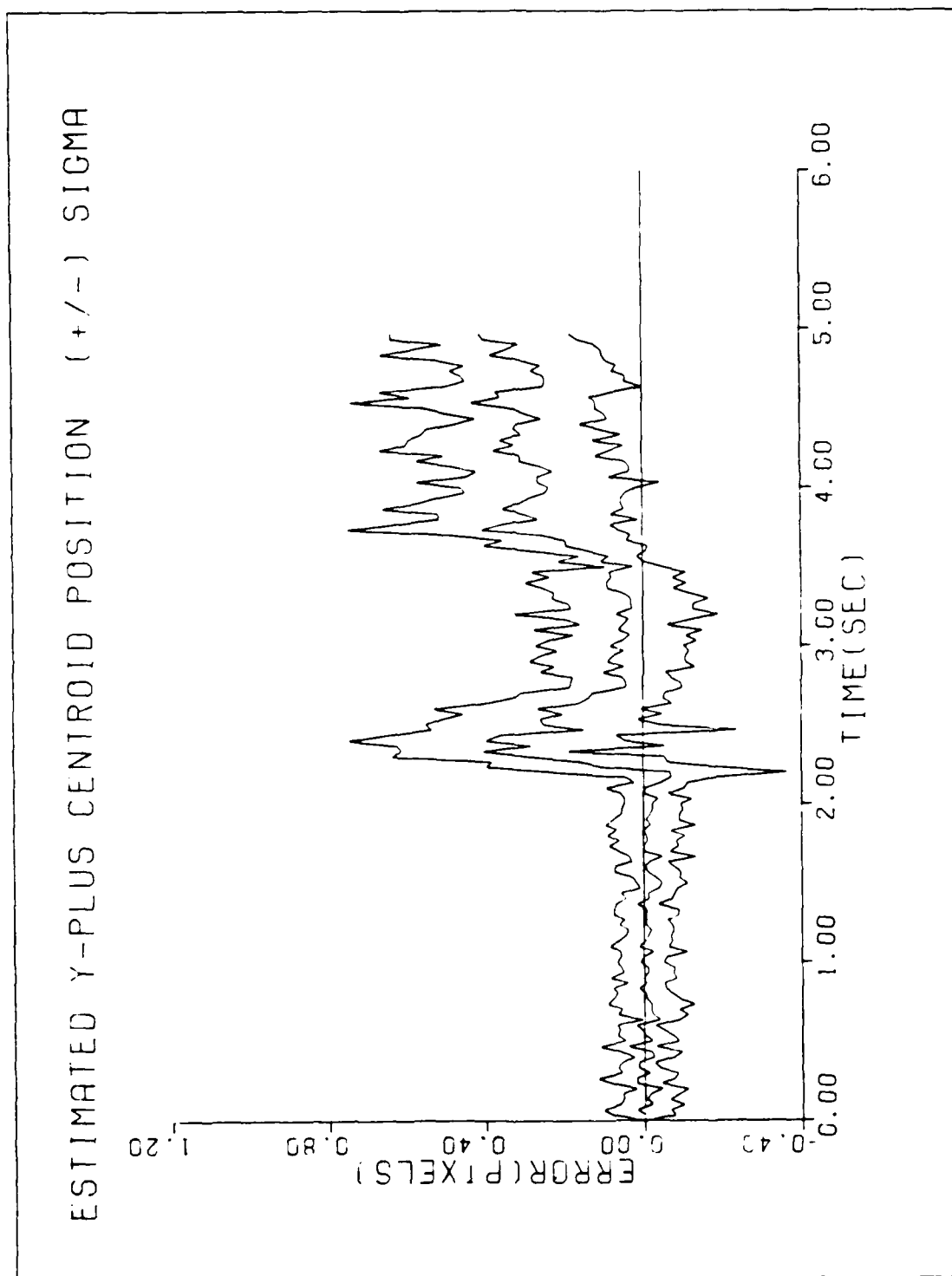


Figure 6-16. GM/MMAF/T6/10-6/QB1

Bibliography

1. Brogan, William L. Modern Control Theory. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1985.
2. D'Azzo, John J. and Constantine H. Houpls. Feedback Control System Analysis and Synthesis. New York: McGraw-Hill, Inc., 1966.
3. Flynn, Patrick M. Alternative Dynamics Models and Multiple Model Filtering for a Short Range Tracker. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1981.
4. Harnly, Douglas A. and Robert L. Jensen. An Adaptive Distributed-Measurement Extended Kalman Filter for a Short Range Tracker. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1978.
5. Harnly, Douglas A., Discussions Concerning Large Space Platform Response Characteristics, June, 1987.
6. Kobel, William and Martin. Distortion-Invariant Pattern Recognition in Non-Random Noise. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1986.
7. Kozemchak, Mark R. Enhanced Image Tracking: Analysis of Two Acceleration Models in Tracking Multiple Hot-Spot Images. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1982.
8. Loving, Phyllis A. Bayesian vs. MAP Multiple Model Adaptive Estimation for Field of View Expansion in Tracking Airborne Targets. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March, 1985.
9. Maybeck, Peter S. Stochastic Models, Estimation, and Control, Volume I. New York: Academic Press, 1979.
10. - - - - - Stochastic Models, Estimation, and Control, Volume II. New York: Academic Press, 1982.

11. - - - and Robert I. Suizu. "Adaptive Tracking of Multiple Hot-Spot Target IR Images, "IEEE Transactions of Aerospace and Electronic Systems, Vol. AES-21, No. 4, 529-539, July, 1985.
12. Mercier, D.E. An Extended Kalman Filter for Use in a Shared Aperture Medium Range Tracker. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1978.
13. Millner, P.P. Enhanced Tracking of Airborne Targets Using a Correlator/Kalman Filter. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1982.
14. Netzer, Allan S. Characteristics of Bayesian Multiple Model Adaptive Estimation for Tracking Airborne Targets MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1985.
15. R & D Associates and Cambridge Research. Space Based Laser (SBL) Structures Final Report and Dynamic Model. Logicon, Albuquerque, NM, 1985.
16. Rogers, Steven K. Enhanced Tracking of Airborne Targets Using Forward Looking Infrared Measurements. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1981.
17. Singletery, James Adaptive Laser Pointing and Tracking Problem. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1980.
18. Suizu, Robert I. Enhanced Tracking of Airborne Targets Using Multiple Model Filtering Techniques for Adaptive Field of View Expansion. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1983.
19. Tobin, David M. A Multiple Model Adaptive Tracking Algorithm for a High Energy Laser Weapon System. MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December, 1986.

VITA

Captain Thomas A. Leeney was born on December 5, 1952 in Evergreen Park, Illinois. He enlisted in the Air Force shortly after graduating from Bogan High School in Chicago. Capt. Leeney received the Bachelor of Science degree in Electrical Engineering from Texas Tech University in May, 1983, via the Airman Education and Commissioning Program, graduating Cum Laude. After graduating from Officers' Training School, in August, 1983, Capt. Leeney was assigned to Aeronautical Systems Division, Wright-Patterson AFB to apply his engineering talents to the acquisition business. In May, 1986, Capt. Leeney was assigned to the Air Force Institute of Technology to pursue the degree of Master of Science in Electrical Engineering.

Permanent Address: 8115 So. Kolmar St.
Chicago, IL 60652

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
GMB No. 2704-0188

1a REPORT SECURITY CLASSIFICATION Unclassified			1b RESTRICTIVE MARKINGS		
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION AVAILABILITY OF REPORT Approved for public release; distribution unlimited		
2b DECLASSIFICATION/DOWNGRADING SCHEDULE					
4 PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/ENG/87D-37			5 MONITORING ORGANIZATION REPORT NUMBER(S)		
6a NAME OF PERFORMING ORGANIZATION School of Engineering		6b OFFICE SYMBOL (If applicable) AFIT/ENG		7a NAME OF MONITORING ORGANIZATION	
6c ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB OH 45433-6583				7b ADDRESS (City, State, and ZIP Code)	
8a NAME OF FUNDING/SPONSORING ORGANIZATION		8b OFFICE SYMBOL (If applicable)		9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c ADDRESS (City, State, and ZIP Code)				10 SOURCE OF FUNDING NUMBERS	
				PROGRAM ELEMENT NO	PROJECT NO
				TASK NO	WORK UNIT ACCESSION NO
11. TITLE (Include Security Classification) A Multiple Model Adaptive Tracking Algorithm Against Airborne Targets					
12. PERSONAL AUTHOR(S) Thomas A. Leeney, B.S., Capt., USAF					
13a. TYPE OF REPORT MS Thesis		13b. TIME COVERED FROM _____ TO _____		14 DATE OF REPORT (Year, Month, Day) 1987 December	
15. PAGE COUNT 122					
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB GROUP			
17	5				
19 ABSTRACT (Continue on reverse if necessary and identify by block number) Thesis Chairman: Dr. Peter S. Maybeck <div style="text-align: right;"><p>Approved for public release: 14W AFR 19047. EXTRA RELEASE AUTHORITY Document Research and Development Air Force Research and Development Wright-Patterson AFB OH 45433</p></div>					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED//UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Peter S. Maybeck			22b. TELEPHONE (Include Area Code) (513) 255-3576		22c. OFFICE SYMBOL AFIT/ENG

Unclassified

This thesis extends the AFIT research directed towards replacing a standard correlation tracker with a Kalman filter bank enhanced correlation tracker in a high energy laser weapon system. Airborne targets are tracked by a Bayesian multiple model adaptive filtering (MMAF) algorithm, which utilizes an array of infrared sensing detectors as the measurement information for two-dimensional position data. Two different target dynamics models are exercised: a linear, Gauss Markov acceleration model, and a nonlinear, constant turn rate model. Performance analyses are accomplished via Monte Carlo simulation techniques. Extending the adaptive potential of the tracking algorithm is of primary emphasis. The effects of bending and vibration of a large space structure on the FLIR's ability to resolve target position is analyzed. Also, a performance comparison/simulation time tradeoff is conducted with the tracking algorithm operating at both 30 Hz and 50 Hz. Sensitivity studies of adaptive responsiveness to varying target trajectories, various filter-assumed correlation times, range to pixel size relationships, and pixel size to filter driving white noise strength relationships are performed. The robustness of the multiple model algorithm is demonstrated by its ability to adapt to scenarios which it had not been previously tuned.

Unclassified

END
DATE
FILMED

4-88
DTIC